



DOCUMENTATION ISG-kernel

Manual **HLI-Documentation V2.20xx**

Short description:
HLI

© Copyright
ISG Industrielle Steuerungstechnik GmbH
STEP, Gropiusplatz 10
D-70563 Stuttgart
All rights reserved
www.isg-stuttgart.de
support@isg-stuttgart.de

Documentation version: 1.22
04/06/2025

Preface

Legal information

This documentation was produced with utmost care. The products and scope of functions described are under continuous development. We reserve the right to revise and amend the documentation at any time and without prior notice.

No claims may be made for products which have already been delivered if such claims are based on the specifications, figures and descriptions contained in this documentation.

Personnel qualifications

This description is solely intended for skilled technicians who were trained in control, automation and drive systems and who are familiar with the applicable standards, the relevant documentation and the machining application.

It is absolutely vital to refer to this documentation, the instructions below and the explanations to carry out installation and commissioning work. Skilled technicians are under the obligation to use the documentation duly published for every installation and commissioning operation.

Skilled technicians must ensure that the application or use of the products described fulfil all safety requirements including all applicable laws, regulations, provisions and standards.

Further information

Links below (DE)

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

or (EN)

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

contains further information on messages generated in the NC kernel, online help, PLC libraries, tools, etc. in addition to the current documentation.

Disclaimer

It is forbidden to make any changes to the software configuration which are not contained in the options described in this documentation.

Trade marks and patents

ISG®, ISG kernel®, ISG virtuos®, ISG dirigent®, TwinStore® and the associated logos are registered and licensed trade marks of ISG Industrielle Steuerungstechnik GmbH.

The use of other trade marks or logos contained in this documentation by third parties may result in a violation of the rights of the respective trade mark owners.

Copyright

© ISG Industrielle Steuerungstechnik GmbH, Stuttgart, Germany.

No parts of this document may be reproduced, transmitted or exploited in any form without prior consent. Non-compliance may result in liability for damages. All rights reserved with regard to the registration of patents, utility models or industrial designs.

General and safety instructions

Icons used and their meanings

This documentation uses the following icons next to the safety instruction and the associated text. Please read the (safety) instructions carefully and comply with them at all times.

Icons in explanatory text

- Indicates an action.
- ⇒ Indicates an action statement.



⚠ DANGER

Acute danger to life!

If you fail to comply with the safety instruction next to this icon, there is immediate danger to human life and health.



⚠ CAUTION

Personal injury and damage to machines!

If you fail to comply with the safety instruction next to this icon, it may result in personal injury or damage to machines.



Attention

Restriction or error

This icon describes restrictions or warns of errors.



Notice

Tips and other notes

This icon indicates information to assist in general understanding or to provide additional information.



Example

General example

Example that clarifies the text.



Programing Example

NC programming example

Programming example (complete NC program or program sequence) of the described function or NC command.



Release Note

Specific version information

Optional or restricted function. The availability of this function depends on the configuration and the scope of the version.

Table of contents

Preface.....	2
General and safety instructions	3
1 Introduction.....	9
1.1 Implementation by High-Level Interface (HLI).....	10
1.2 Organisation of the HLI	11
1.2.1 Data of a channel\axis	13
1.3 Status and display information	13
1.4 Control commands	14
1.4.1 Control unit with usage check	15
1.5 Operation with and without PLC.....	16
2 Axes	18
2.1 Definition of axes.....	18
2.2 Definition of coordinate systems	18
2.3 Description of the axis-specific interface	20
2.3.1 External axis commanding	20
2.3.2 Axis identification	21
2.3.3 Axis positions	23
2.3.3.1 Axis positions in the PCS.....	23
2.3.3.2 Axis positions in the ACS.....	25
2.3.4 Position of tool centre point in MCS.....	27
2.3.5 Status information of an axis.....	27
2.3.5.1 Drive of an axis	35
2.3.5.2 Compensation of an axis	41
2.3.5.3 Measurement	43
2.3.5.4 Link to ADS	44
2.3.6 Axis control commands.....	45
2.3.6.1 Commanding axis couplings	62
2.3.6.2 Distance control	66
2.3.7 Control commands of a drive	72
2.3.8 Measuring with external measuring hardware	74
3 Spindle.....	77
3.1 Introduction	77
3.2 Description of the spindle-specific interface	77
3.2.1 Speeds of a spindle	77
3.2.2 Positions of a spindle	78
3.2.3 Status information of a spindle.....	78
3.2.4 Control commands of a spindle	81
3.2.5 External spindle command.....	83
3.2.5.1 Control unit of external spindle command	83
3.2.5.2 User data for external spindle commands	83
4 Channel	87
4.1 Introduction	87
4.2 Description of the channel-specific interface.....	88

4.2.1	Status information of a channel	88
4.2.1.1	Status information on tool orientation	102
4.2.1.2	Diagnosis upload	103
4.2.2	Control commands of a channel	105
5	PLC	129
5.1	Control commands to PLC	129
5.1.1	Reset.....	129
5.1.2	Block search	131
6	Technology processes.....	134
6.1	Introduction	134
6.2	Management of technology functions	134
6.3	Elements to manage axis-specific technology control units.....	135
6.3.1	Block-by-block synchronisation (default synchronisation)	135
6.3.2	Cross-block synchronisation	135
6.4	Elements for managing channel-specific technology control units.....	136
6.4.1	Block-by-block synchronisation (default synchronisation)	136
6.4.2	Cross-block synchronisation	136
6.5	Data of a technology control unit.....	137
6.5.1	Data of an axis-specific technology control unit.....	137
6.5.2	Data of a channel-specific technology control unit.....	139
6.6	Data of the technology functions.....	142
6.6.1	Data of the M/H function	142
6.6.2	Data of the S function	144
6.6.3	Data of the T function.....	146
6.6.3.1	User data of tool identification	147
7	External variables / V.E variables	149
8	Processing types.....	150
8.1	State graph of operation modes.....	150
8.1.1	States of operation mode: Automatic mode.....	152
8.1.2	States of operation mode: Manual block.....	152
8.1.3	States of operation mode: Manual operation mode	153
8.1.4	States of operation mode: Homing	153
8.2	Control commands/status information for operation modes.....	153
8.2.1	Control Unit.....	154
8.2.2	User data	155
8.2.2.1	Requested and commanded user data.....	155
8.2.2.2	Status information	159
9	Manual operation mode	160
9.1	Status information of manual mode	161
9.2	Control commands of manual mode	164
9.2.1	Activating control elements for manual mode	164
9.2.1.1	User data on activation	166
9.2.2	Parametrising of manual mode	167
9.2.2.1	Continuous jog mode (by button press).....	168
9.2.2.1.1	Control Unit.....	168

9.2.2.1.2	User data	169
9.2.2.2	Incremental jog mode (by button press)	170
9.2.2.2.1	Control Unit.....	170
9.2.2.2.2	User data	171
9.2.2.3	Handwheel mode.....	173
9.2.2.3.1	Control Unit.....	173
9.2.2.3.2	User data	173
9.2.3	Control elements for manual mode.....	174
9.2.3.1	Enforcing a button press.....	174
9.2.3.1.1	Control Unit.....	175
9.2.3.1.2	User data	177
9.2.3.2	Rapid traverse velocity during path motion.....	178
9.2.3.2.1	Control Unit.....	179
9.2.3.2.2	User data	179
9.2.3.3	Handwheel increments	180
10	Safety system	182
10.1	Channel-specific interface.....	182
10.1.1	Watchdog mechanism	182
11	Management.....	186
11.1	Channel-specific interface.....	186
11.2	Axis-specific interface	189
12	Error messages	191
12.1	Management data of an error message	191
12.2	User data of an error message	191
12.2.1	Error message content, body nc program.....	196
12.2.2	Error message content, body machine data	197
12.2.3	Error message content, body communication.....	198
12.2.4	Error message content, body RAM disk	199
12.2.5	Error message content, body file	200
12.2.6	Error message content, body interpreted file list.....	201
12.2.7	Error message content, body binary list.....	202
12.2.8	Error message content, body global channel manager	203
12.2.9	Additional error information value 1 - 5.....	205
12.2.9.1	Value range of variable 'Type' of additional error information	206
12.2.9.2	Value range of 'Dimension' variable of the additional error information	207
12.2.9.3	Value range of 'Significance' variable of additional error information	208
13	Messages	209
13.1	Control Unit	209
13.2	User data.....	210
13.2.1	Requested and commanded user data.....	210
14	Implementation as PLC library.....	211
14.1	Access to HLI	211
14.1.1	PLC system TwinCAT	211
14.2	Function blocks within the PLC library	211
14.2.1	Overview of the PLCopen FBs.....	211
14.2.2	ISG HLI Interface	212

15	Programming examples	214
15.1	PLC main program frame	214
15.1.1	Initialisation function UserInitialisations()	214
	Keyword index	216
16	Appendix	226
16.1	Suggestions, corrections and the latest documentation.....	226

List of figures

Fig. 1:	CNC-PLC interface	9
Fig. 2:	Structuring of the HLI	11
Fig. 3:	Channel-specific memory area	12
Fig. 4:	Axis-specific/spindle-specific memory area	12
Fig. 5:	Transfer of status information	13
Fig. 6:	Application of an MC control unit	14
Fig. 7:	Interaction, MC control unit and PLC	16
Fig. 8:	Coordinate systems	19
Fig. 9:	Position values in PC and AC	20
Fig. 10:	Interaction of control units and SERCOS control or status words	49
Fig. 11:	Example signal profile of measurement with external hardware	76
Fig. 12:	Vectors of the tool coordinate system	102
Fig. 13:	Interaction between feedhold and NC channel stop	111
Fig. 14:	Time sequence of feedhold and NC channel stop	112
Fig. 15:	Interaction of input disable and NC channel stop	120
Fig. 16:	Time sequence of input disable and NC channel stop	121
Fig. 17:	Interaction between BOOLEAN-MC control unit and PLC	131
Fig. 18:	Interaction between BOOLEAN-MC control unit and PLC	133
Fig. 19:	State graph of an operation mode	151
Fig. 20:	Control elements and assignment	160
Fig. 21:	Manual mode state transitions	161
Fig. 22:	Priorities for watchdog mechanism (example TwinCAT 2)	185
Fig. 23:	Management data of the channel-specific interface	186
Fig. 24:	FB parameters	212

1

Introduction

Extensive data volumes are exchanged between the CNC and the PLC. For example, this comprises

- Commands from the NC program of the CNC to the PLC (e.g. technology commands, such as M, S, T and H commands etc.).
- Acknowledgements of technology commands by the PLC.
- Display data of the CNC (e.g. current axis positions, current and programmed path velocity etc.).
- Jobs from the PLC to the CNC (e.g. mode switchover, set feedhold etc.).
- Jobs from the GUI to the CNC which can be verified and, if applicable, denied by the PLC (e.g. mode switchover, set feedhold etc.).

The interface between CNC and PLC is outlined in the overview diagram below.

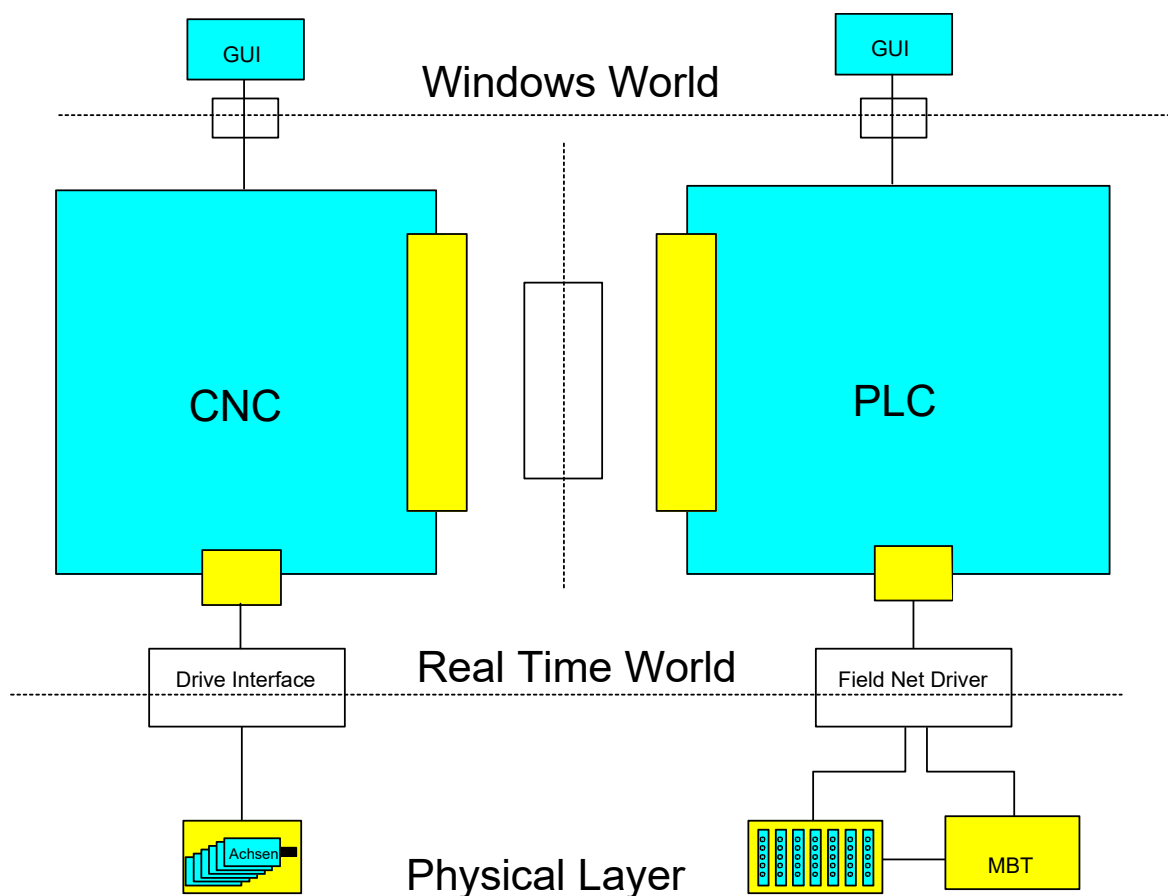


Fig. 1: CNC-PLC interface

This documentation describes the structure of this interface between CNC and PLC referred to below as High-Level Interface (HLI for short).

The term “High-Level Interface”, in this context, refers to the structure of the interface with complex data structures and handshake variables. This has to be seen in contrast to a data interface, which is considered to be at a low level of abstraction (Low-Level Interface).



Attention

Safety restriction for the use of the High Level Interface:

The values transferred between the PLC and TwinCAT CNC can be read and changed by all local users

In order to protect the system against unauthorised access, use the “IPC Security Guide” as a guideline. This will support you in the management of security risks when using Beckhoff products.

Follow the links to the „IPC Security Guide“ for further measures:

German: https://infosys.beckhoff.com/content/1031/ipc_security/index.html

English: https://infosys.beckhoff.com/content/1033/ipc_security/index.html

1.1

Implementation by High-Level Interface (HLI)

A memory area is created as shared memory which both the CNC and the PLC can access in order to implement data exchange between CNC and PLC. This memory area is referred to as High-Level Interface (HLI).

In order to allow this data access, the relevant views from CNC and PLC onto the memory area must be identical. Data consistency is ensured by means of suitable data exchange mechanisms.

A library containing the structure of the HLI in structured text is provided for simple programming at the PLC end in IEC1131-3. This library is valid in each case for the current HLI format and the current status of the CNC and may change if a version update is implemented. Accordingly, the PLC must be recompiled and loaded if there is new library or if a new CNC is used.

1.2

Organisation of the HLI

As opposed to the usual simple structure of the data to be exchanged in input/output data with PLC applications, the HLI contains complex structures. They reflect the logical structuring of the CNC into channels, axes and platform data.

The logical structure of the HLI is shown in the diagram below:

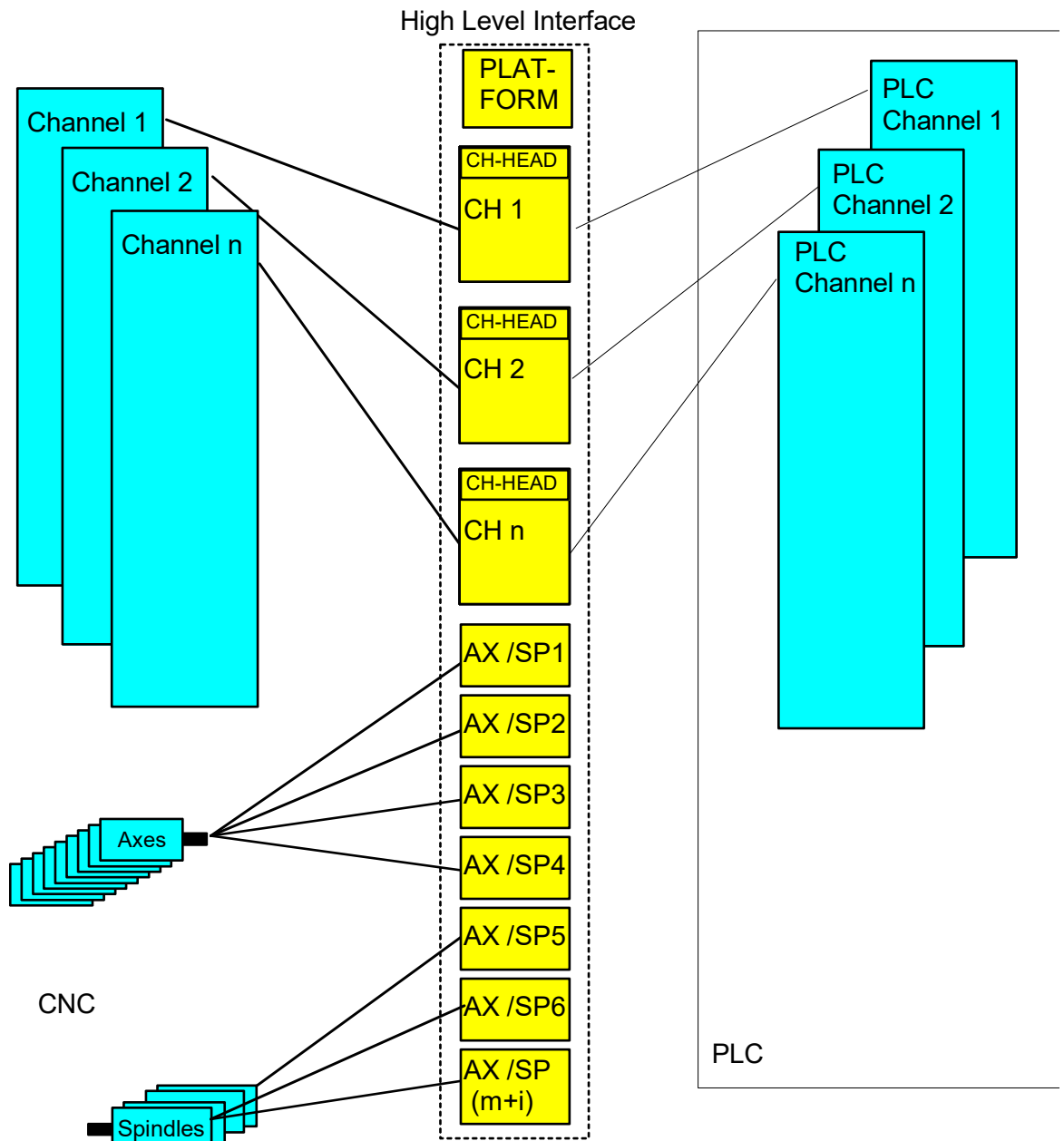


Fig. 2: Structuring of the HLI

The structuring into channel-specific and axis-specific data areas is clearly shown. The channel-specific and axis-specific memory areas on the HLI feature the same logical structure.

The relevant memory areas are subdivided into

- a header area with management data
- and a user data area with status information, control commands and technology data.

The channel/axis-specific memory areas are shown magnified in the diagram below:

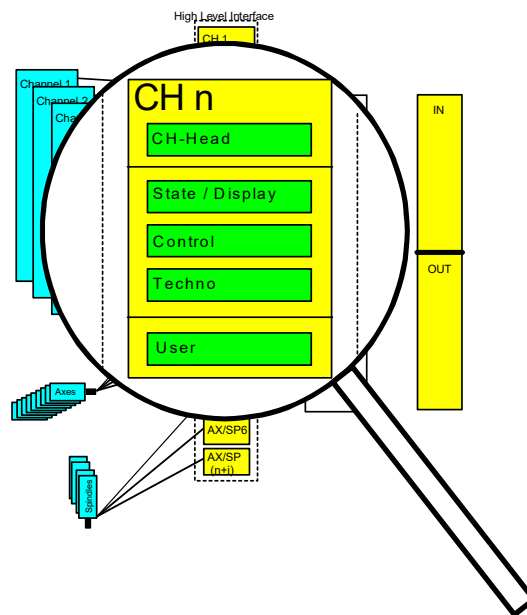


Fig. 3: Channel-specific memory area

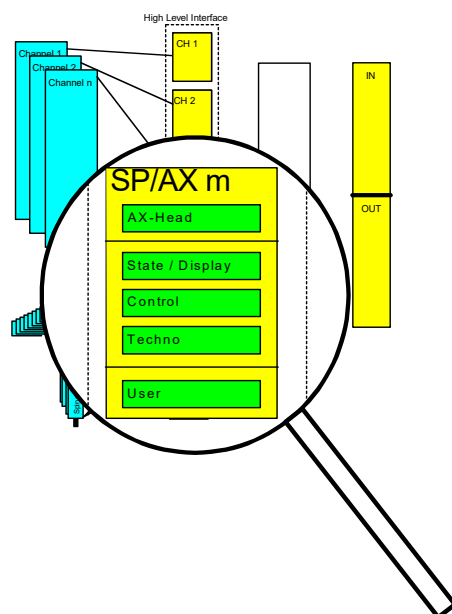


Fig. 4: Axis-specific/spindle-specific memory area

1.2.1 Data of a channelaxis

A distinction is made between the following data areas, whereby channels and axes may be considered the same:

The header areas contain:

- management data such as version information and log-on information

The user data areas contain:

- status and display information (CNC → PLC),
- control interfaces (PLC → CNC),
- technology areas (CNC → PLC and PLC → CNC),
- and any application-specific data.

Specific data, such as status information, is refreshed cyclically by the CNC and can be read by the PLC if needed. However, M functions, so-called usage information, must be read by the PLC. For this purpose, the interface contains suitable mechanisms to ensure that no data is lost and that the order of the data is retained.

1.3 Status and display information

The status information is transferred unidirectionally from the CNC to the PLC. This data is constantly refreshed by the CNC and can be read by the PLC if needed. The status information is updated without informing the PLC. Updating is not protected by a handshake protocol or semaphore mechanism.

With long PLC cycle times, the PLC does not record each short-term change under certain circumstances but is only informed of the current status.

Status information is exchanged on the HLI via memory. The transfer direction is defined for each status information item. The status data is transferred on the HLI as far as possible as single elementary data items (byte, Boolean, integer etc.).

The status information is combined depending on the CNC-internal structure to enhance structuring.

The diagram below shows the internal structure of the CNC in simplified form:

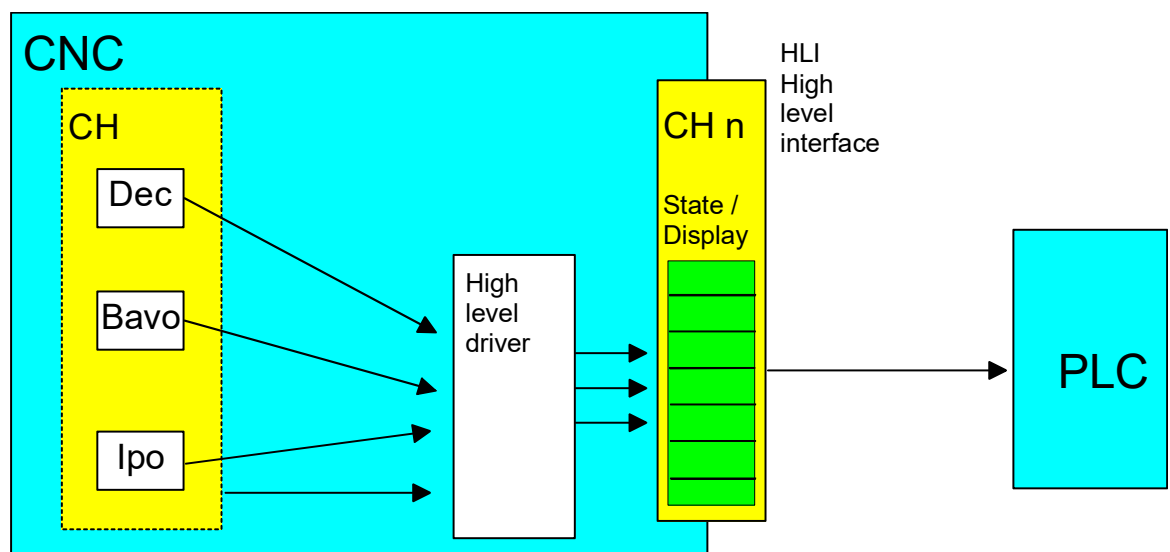


Fig. 5: Transfer of status information

The CNC is subdivided into the modules:

- **Dec** (Decoder): Module for decoding the NC program. This module converts the CNC program into an internal control data format and executes a number of different calculations, such as parameter calculation etc. Timewise, the decoder runs ahead of actual processing and under certain circumstances may be several hundred NC blocks ahead of current processing.
- **Bavo** (path preparation, look-ahead): Module to calculate dynamic limit values based on the set parameters. It detects corners and triggers geometry changes. The Bavo also runs asynchronously with current processing and under certain circumstances may be several hundred NC blocks ahead of current processing.
- **Ipo** (interpolator): Module to generate individual positions for the current cycle, allowing for programmed and maximum velocities etc. . The interpolator represents the current state of the machine.

1.4 Control commands

Control commands are transferred both from the CNC to the PLC and also in the opposite direction.

If a user interface (GUI) communicates via CNC communication objects, there is an option to redirect via the PLC each command which can be operated both by the GUI and the PLC. The PLC then makes the decision regarding the extent to which the GUI command may be transferred to the CNC.

A so-called control unit is created on the HLI for each control command. The control units are differentiated and named based on the target action.

- Control units to influence the CNC are referred to as MC control units (LC acts on MC).
- Control units to influence the PLC are referred to as LC control units (MC acts on LC).

The figure below clearly shows the implemented interaction between users of a control system by means of an MC control unit.

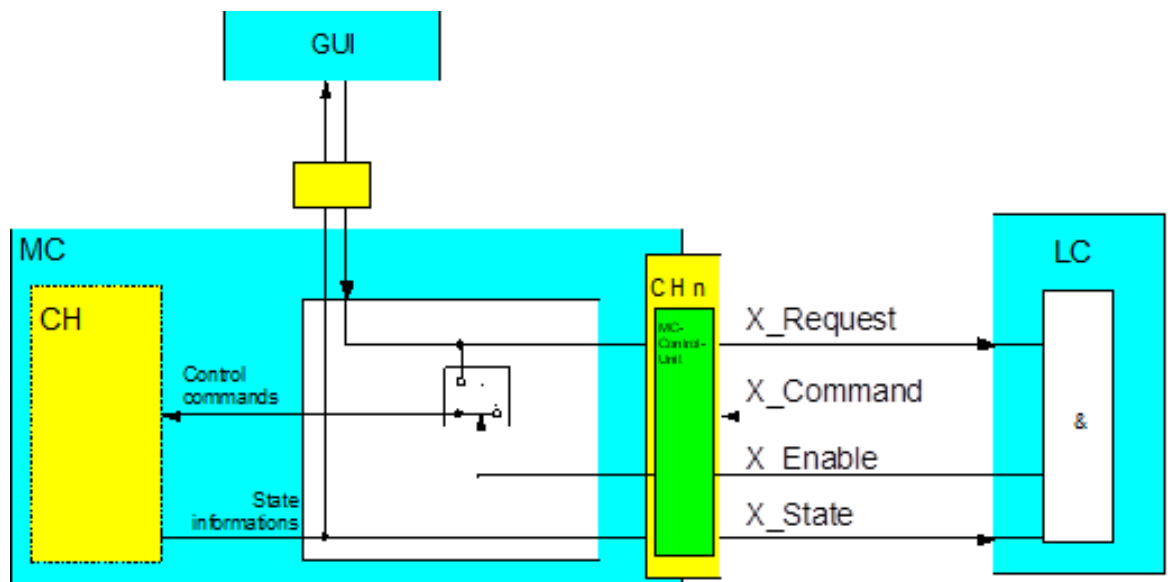


Fig. 6: Application of an MC control unit

If commands from the GUI are redirected via the PLC, the PLC must assign the element **X_Enable** the value TRUE. The corresponding GUI commands are then set to the element **X_Request** in the MC control unit. This means that the PLC has the option of allowing or rejecting this request from the GUI. When a GUI command is permitted by the PLC, the PLC must copy the command from the element **X_Request** to the element **X_Command**.

Similarly, the PLC can directly send control commands to the CNC even without a prior job from a GUI by writing element **X_Command**.

The element **X_State** is used to check the success of a command. The CNC saves the status corresponding to the command in this element.

An **MC control unit** has the following structure:

```

TYPE MCControlUnit:
STRUCT
    X_Request      : < DATATYPE A >; (* data commanded by GUI *)
    X_Enable       : HLI_BOOLEAN;      (* PLC operates this command *)
    X_Command      : < DATATYPE A >; (* data commanded by the PLC *)
    X_State        : < DATATYPE B >; (* CNC feedback *)
END_STRUCT
END_TYPE

```

Note:

- < DATA TYPE A > and < DATA TYPE B > may be identical.
- < DATA TYPE A > and < DATA TYPE B > may also be complex data structures besides the standard data types (e.g. BOOL, INT, UINT...).

1.4.1

Control unit with usage check

If the requested or commanded data is not applied statically and if an individual change should be transferred at this point, the data of the corresponding control unit is provided with an additional usage check.

```

TYPE MCControlUnit:
STRUCT
    X_Request      : < DATATYPE A >; (* data commanded by GUI *)
    X_Enable       : HLI_BOOLEAN;      (* PLC operates this command *)
    X_Command      : < DATATYPE A >; (* data commanded by the PLC *)
    X_RequestSemaphore : HLI_BOOLEAN;      (* Request valid *)
    X_CommandSemaphore : HLI_BOOLEAN;      (* Command valid *)
    X_State        : < DATATYPE B >; (* CNC feedback *)
END_STRUCT
END_TYPE

```

Example:

The CNC accepts the commanded data if the command semaphore has the value TRUE and sets this element to the value FALSE after complete acceptance of the data.

The PLC can write command data if the command semaphore has the value FALSE. The PLC sets this element to TRUE when all commanded data is written.

The CNC writes the data requested by the GUI if the request semaphore is FALSE and then sets this element to TRUE.

The PLC reads the data requested by the GUI if the request semaphore is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.

1.5 Operation with and without PLC

Machine start-up can also be performed with partially implemented PLC functionality. The CNC-side driver of the HLI therefore contains mechanisms for the internal simulation of interactions between PLC and CNC.

Example:

The axis should be run in with a rudimentary PLC. The PLC sets only the Axis Enables; other functions are not yet implemented in the PLC.

In order to switch operation modes or perform a reset anyway, the HLI driver simulates the required acknowledgements of the PLC so that the required action can still be performed.

In order to achieve this behaviour and still ensure reliable operation, the PLC issues a corresponding information item to the CNC for each function covered by the PLC, a so-called “present” marker. “Present” means that the PLC claims control of the interface and sends the required acknowledgements.

The PLC must then inform the CNC that it exists, once globally in the HLI. In addition all control commands are enabled individually.

This means that three scenarios are conceivable:

- CNC without PLC
- CNC with PLC which does not use all elements of the HLI
- CNC with PLC which uses all elements of the HLI

The housekeeping sections of each channel or each axis has this “present” element by which the PLC assumes responsibility for this interface.

In addition, each housekeeping unit of a control command, i.e. the control unit, contains an element which the PLC uses to inform the CNC that it supports this command. The **X_Enable** element means at the same time that, if the corresponding command is issued via the GUI, it is routed via the PLC and reaches the CNC from there. The CNC can then no longer distinguish whether a command arrives from the PLC or the GUI.

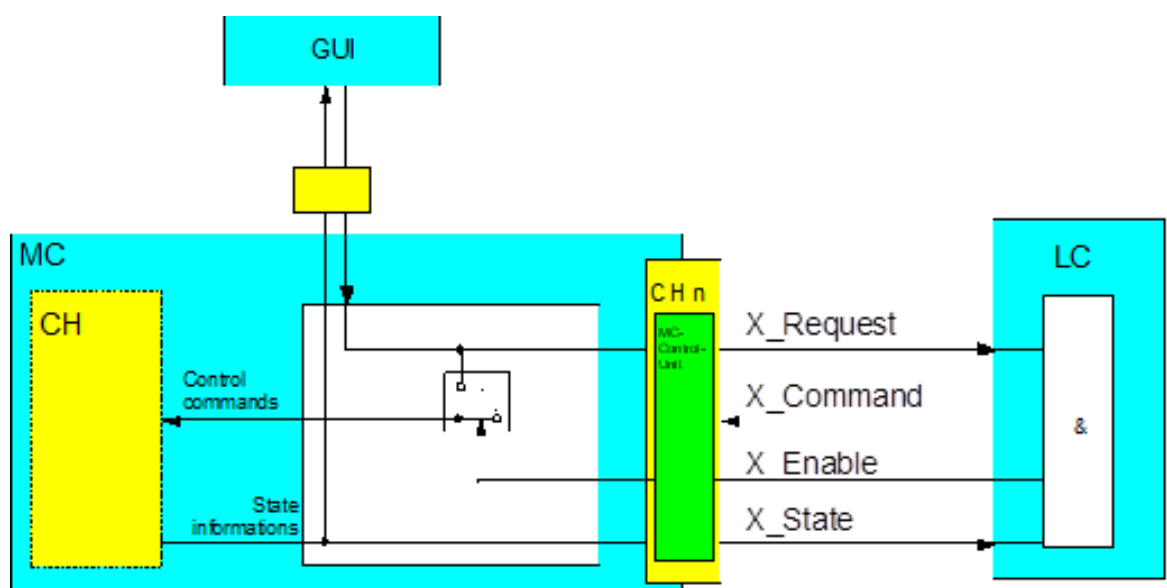


Fig. 7: Interaction, MC control unit and PLC

The connection of the GUI via the CNC is not absolutely necessary. Similarly, an interface can communicate directly with the PLC and the additionally provided communication channel GUI -> CNC -> PLC is not used in such cases.

2 Axes

2.1 Definition of axes

A distinction can be made between various types of axes in a control system. One practical classification is between programmable, logical and physical axes.

A **physical axis** corresponds to a real axis on the machine (translatory or rotary axis). In a control unit a physical axis is represented by software by means of a **logical axis** (1:1 mapping).

A logical axis is the unified representation of an axis in the axis coordinate system (acs). A logical axis in the control unit provides all the relevant information required for the related physical axis. From the point of view of the CNC, each logical axis has its own parameter set and its own interface to the PLC.

A **programmable axis** is an axis which can be programmed or commanded by the user in the part program or MDI mode. A programmable axis either corresponds directly to a logical/physical axis (1:1 mapping) or is mapped by kinematic transformation onto logical/physical axes (1:n mapping).

In multi-channel operation, a logical axis may be swapped between channels ("axis swap"). In this case, the same logical axis may occur in different attributes on the NC channels from the point of view of the programmer, e.g. as a programmable spindle on one channel (speed preset, speed-controlled) and as a programmable C axis on another channel (position preset, position-controlled).

Programmable, logical and physical axes are generally identical (1: 1 mapping) for simple machines (e.g. a 3-axis milling machine with Cartesian linear axes X, Y and Z). However, with complex machine kinematics or industrial robots, kinematic transformation of programmed to logical/physical axes is required (1:n mapping).

Axes are programmed in the CNC program and moved by the CNC. Alternatively, specific motions may also be triggered directly by the PLC.

The PLC interface for axes has a similar structure as the interface for channels. M functions configured as axis-specific M functions also appear on the axis-specific interface and must be acknowledged by this interface. Similarly, this interface contains axis enables and status information for this axis.

2.2 Definition of coordinate systems

Allowance must be made for various coordinate systems owing to the structure of a machine and its machining programs. The totality of a machine is represented by the reference coordinate system in world coordinates. In turn, the individual axes themselves define coordinate systems, the workpiece and the tool.

The two different coordinate systems mainly used are described below.

Axis coordinate system acs

Each axis has its own coordinate system. Each axis is mounted either on the machine base or on another axis. This means that the machine base or the corresponding axis forms the basis. The axis coordinate system of an axis is fixed with respect to the mounting point of this axis.

Part program coordinate system pcs

This coordinate system is used in the Geometry Description with the DIN 66025 programming language. The data in a part program constitutes program coordinates. Exceptions are G functions which refer to direct axis coordinates.

Other coordinate system designations are listed for the purpose of completeness.

Machine coordinate system mcs

The machine coordinate system represents an abstract coordinate system. It is not bound to a fixed point of the machine. All other coordinate systems refer to this coordinate system.

Workpiece coordinate system wcs

This coordinate system is fixed to a fixed point of the workpiece. The workpiece description by coordinate information refers to this system.

Tool coordinate system tcs

The tool coordinate system has its origin at the clamping point of the tool. Tool information (geometry) refers to this system. Length compensation is therefore specified in tool coordinates. On Cartesian machines, the Z axis may coincide with length compensation.

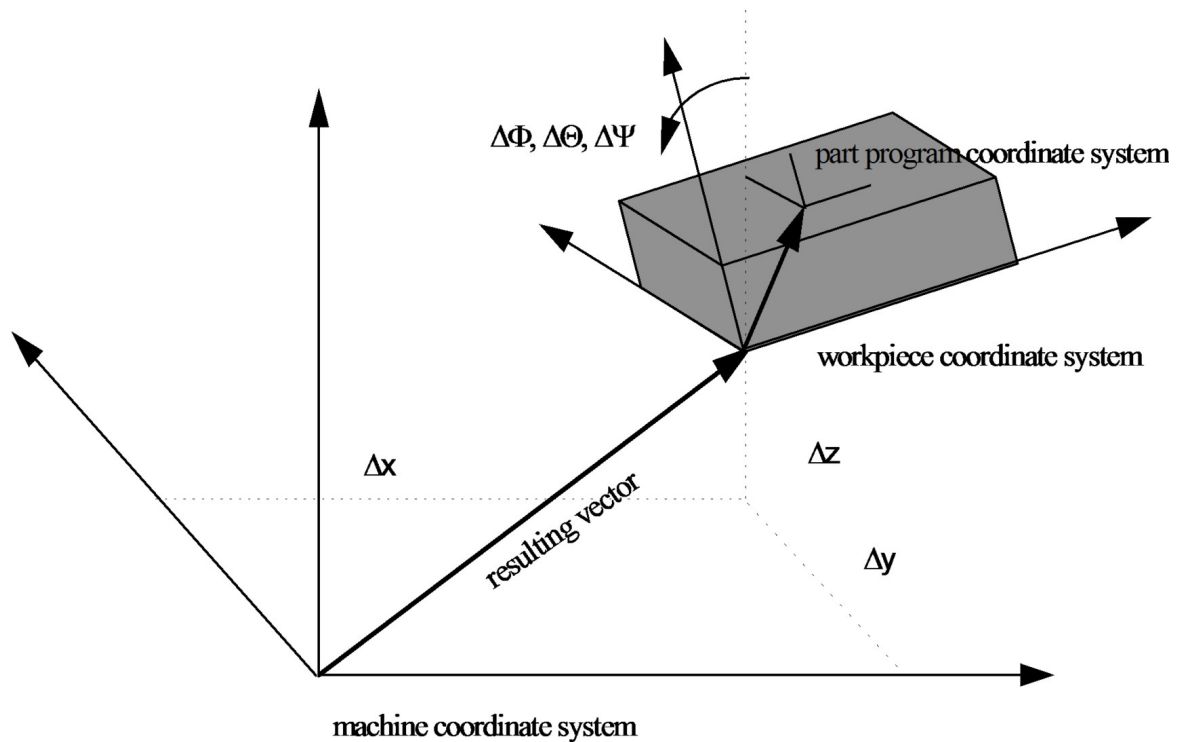


Fig. 8: Coordinate systems

Consequently, data or variables described below always refer to a specific coordinate system. Three coordinate systems are shown in the figure above. The offsets depicted Δx , Δy and Δz and the orientations $\Delta \Phi$, $\Delta \Theta$ and $\Delta \Psi$ represent the transformation parameters from mcs to wcs. The orientation angles are Euler angles.

Naming must be supplemented by means of an additional suffix in order to allow for the time aspect. The figure below shows the machining direction of a machining block. The **end position** represents the programmed value; the **active position** is the current value of the interpolator; and the **current position** is the actual position including the control error.

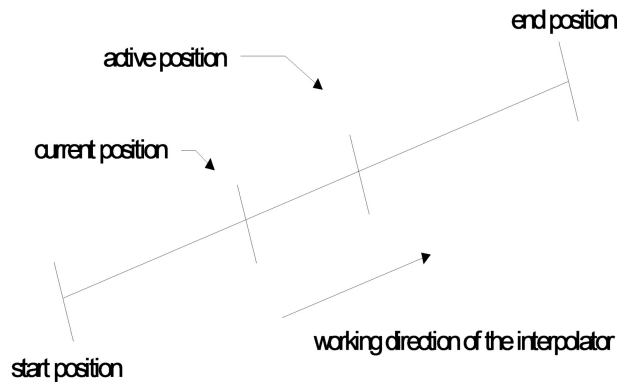


Fig. 9: Position values in PC and AC

2.3 Description of the axis-specific interface

2.3.1 External axis commanding

Activating external position or velocity command values, axis	
Description	Specifying velocity or position command values by the SPS effective in addition to the interpolator. No monitoring takes place of transferred values for compliance with the dynamic axis limits. To activate this interface, set the parameter P-AXIS-00732 to 1.
Data type	MC_CONTROL_ADD_CMD_VALUE_UNIT, see description of Control Unit
Special features	When this interface is used, the axis positions in the interpolator and decoder are permanently offset. A repeated synchronisation of axis positions is executed, e.g. at program start, after homing, after measuring or by the NC command #CHANNEL INIT or #SET DEC LR SOLL (old syntax). If synchronisation should not take place (the offset remains as static offset), set the parameter P-AXIS-00322 to the value 1.
Access	PLC writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlAddCmdValueUnit_AddCmdValue
Commanded values	
ST element	.HLIAddCmdValue_Command
Data type	HLIAddCmdValueData [► 20]
Activation	
ST element	.X_Enable

External command value, axis	
Description	Transferring additional position or velocity command values. On activation both values are effective at the same time.
Data type	HLIAddCmdValueData
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlAddCmdValueUnit_AddCmd-Value. HLIAddCmdValue_Command
Data structure elements	
ST element	.D_AddPosValue
Special features	Absolute value
Data type	DINT
Unit	0,1 µm
Access	PLC writes
ST element	.D_AddSpeedValue
Data type	DINT
Unit	1 µm/s
Access	PLC writes

2.3.2 Axis identification

Axis name (PCS)	
Description	Name of the logical axis used for the current reference in the current automatic program / manual block (e.g. X, Y, Z). This may be changed by default when the channel (SDA-MDS list) is programmed or dynamically in the NC program by means of a swap command.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. HLIAxeName_Data.STR_Zeichen
Data type	ARRAY[1..GCW_250_HLIAchsNameLaenge] OF BYTE
Access	PLC reads

Axis number (PCS)	
Description	If an axis is assigned to a channel, the logical number of this axis is indicated here. The logical axis number is unique in the entire system. The logical axis number is defined arbitrarily in the axis parameter lists and is required to identify the axis in case of an axis swap, among other things (e.g. #CALL AX [X, 1, 0]).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. W_LogAchsNr pAC[axis_idx]^addr^.Statelpo_Data. W_LogAxeNr
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]
Access	PLC reads

Axis spindle (ACS)	
Description	Indicates whether the axis is currently interpolated by the path interpolator (e.g. thread drilling, tapping) or the BF spindle.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.Statelpo_Data. X_SpindleAxis
Data type	BOOL
Value range	[TRUE = the axis is traversed by the BF spindle, FALSE]
Access	PLC reads

Axis type (PCS)	
Description	Type of axis
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. W_AxisType pAC[axis_idx]^addr^.Statelpo_Data. W_Type
Data type	UINT
Value range	1 = Translator, 2 = Rotator, 4 = Spindel
Access	PLC reads

Channel number of the axis	
Description	Number of the channel by which the axis is currently commanded. == 0: Axis is currently not on a channel or spindle is not processing any channel command. != 0: Axis belongs to a channel or spindle is executing a channel command.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.W_NCChannel pAC[axis_idx]^addr^.StateIpo_Data.W_NCChannel
Data type	UINT
Value range	[1, HLI_SYS_CHNMAX]
Access	PLC reads

2.3.3 Axis positions

2.3.3.1 Axis positions in the PCS

Target position (PCS)	
Description	Target position of the current NC block.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_CmdPosition
Data type	DINT
Access	PLC reads

Command position (PCS)	
Description	Position preset in the current cycle as setpoint.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_ActPosition
Data type	DINT
Access	PLC reads

Actual position (PCS)	
Description	Actual ACS position converted in the PCS.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^^.addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. D_CurrentPosition
Data type	DINT
Access	PLC reads

Manual mode offset (PCS)	
Description	Current manual mode offset.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^^.addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. D_ManOffset
Data type	DINT
Access	PLC reads

Total offset (PCS)	
Description	Sum of all active offsets TotalOffset = ACS - offsets + tool
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_TotalOffset
Data type	DINT
Access	PLC reads
Special features	Offsets are included in the total as negative values. For 2.5 D machining: TotalOffset = - Offset_G92 - Offset_NP - ... + WZ_Achsversätze Calculation of PCS coordinates based on ACS coordinates: PCS = ACS + TotalOffset Example program: <pre>%total_offset N00 X100 (ACS = 100mm) N10 G92 X11 (total_offset = -11mm) N20 X200 (ACS = 211mm) N30 D1 (total_offset = 64mm,) (bei wz[1].ax_ersatz[0] 750000[0.1µm]) N40 V.G.WZ_AKT.V[0]=55 (total_offset = 44mm) N30 X300 (-> ACS coordinate = 256 mm) M30</pre>

2.3.3.2 Axis positions in the ACS

Target position (ACS)	
Description	Target position in the current NC block, ACS. This represents the target position of the program coordinate system referred to the axes. It is valid only as long as no transformation is active. Currently, the target position is not transformed back onto the axes.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^addr^.StateLR_Data.D_EndPositionACS
Data type	DINT
Access	PLC reads

Actual position (ACS))	
Description	Actual position of the current cycle in the axis coordinate system
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^^.addr^.StateLR_Data. D_CurrentPositionACS
Data type	DINT
Access	PLC reads

Command position (ACS)	
Description	Command position of current cycle in the axis coordinate system
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^^.addr^.StateLR_Data. D_ActivePositionACS
Data type	DINT
Access	PLC reads

Position offset	
Description	Display of the current offset of the absolute position between the position controller axis (real physical axis) and the interpolator axis (logical axis) currently linked in the axis coordinate system. The display here shows an offset between the position controller position and the interpolator position caused by the use of functions such as “jog of path” (see [FCT-C15]). The offset displayed contains no offsets caused by zero offsets, for example.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data. D_PositionOffsetPhysicalLogicalACS
Data type	DINT
Value range	0.1 µm
Access	PLC is reading

2.3.4 Position of tool centre point in MCS

Tool centre point position (MCS)	
Description	Command position of tool centre point in machine coordinate system MCS. The value is refreshed in each interpolation cycle.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^^.addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. D_W0Position
Data type	DINT
Access	PLC reads

2.3.5 Status information of an axis

Axis state (PCS)			
Description	Axis state, PCS		
Signal flow	CNC → PLC		
ST path	pMC[channel_idx]^^.addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. W_State		
Data type	UINT		
Value range	Value	PLC constant	Meaning
	1	GCW_250_HLIAxeReady	The axis is ready and moves according to the specified command values after a command.
	3	GCW_250_HLIAxeActive	The axis is currently moved by the CNC due to an NC command or manual mode.
	5	GCW_250_HLIAxeHold	Axis cannot be moved by the CNC since feed-hold is active.
Access	PLC reads		

Axis state (ACS)			
Description	Axis state, ACS		
Signal flow	CNC → PLC		
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.W_AxisState		
Data type	UINT		
Value range	Value	PLC constant	Meaning
	1	GCW_250_HLIAxeReady	The axis is ready and moves according to the specified command values after a command.
	3	GCW_250_HLIAxeActive	The axis is currently moved by the CNC due to an NC command or manual mode.
	5	GCW_250_HLIAxeHold	The CNC cannot move the axis because an external signal is set, such as feedhold or tracking mode, or the required drive enables are missing.
	7	GCW_250_HLIAxeError	After an error (in the drive or CNC, e.g. a software limit switch violation) the axis is in error state. Commanding a new motion is only possible after a CNC reset.
Access	PLC reads		
Special features	Even if an axis is not moved in the PCS, a corresponding Cartesian or kinematic transformation may nevertheless execute a motion of the physical axis. Example: 90° rotation about Z; Y is moved if X is programmed.		

Distance to go (PCS)	
Description	Distance to go in the current NC block, difference between target position and command position.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pMC[channel_idx]^^.addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].D_DistToGo
Data type	DINT
Access	PLC reads

Homing executed (PCS)	
Description	The axis completed homing successfully and is now referenced.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx]. X_HomingDone
Data type	BOOL
Value range	[TRUE = axis is referenced, FALSE]
Access	PLC reads

Homing executed (ACS)	
Description	The axis completed homing successfully and is now referenced.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data. X_HomingDone
Data type	BOOL
Value range	[TRUE = axis is referenced, FALSE]
Access	PLC reads

Position lag	
Description	Current position lag of the axis, difference between command position and actual position.
Signal flow	CNC → PLC
Unit	0,1 µm
ST path	pAC[axis_idx]^addr^.StateLR_Data. D_FollowingError
Data type	DINT
Access	PLC reads

Axis in control window	
Description	The axis is located in the control window, i.e. the value of the position lag is less than the currently effective position window (P-AXIS-00236 or P-AXIS-00472).
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data. X_InWindow
Data type	BOOL
Value range	[TRUE = axis is located in control window, FALSE]
Access	PLC reads

Axis in position	
Description	The axis is located in position, i.e. the control window is reached (see above) and the axis is not interpolated.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_InPosition
Data type	BOOL
Value range	[TRUE = Axis in position, FALSE]
Access	PLC reads

Axis is moved	
Description	The axis is currently co-used for the programmed path motion.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_IsMoved
Data type	BOOL
Value range	[TRUE = Axis is moved, FALSE]
Access	PLC reads

Axis traverses	
Description	The axis traverses, i.e. a command value is generated for this axis for the current interpolation.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_IsMoving
Data type	BOOL
Value range	[TRUE = Axis is traversed, FALSE]
Access	PLC reads
Special features	If an axis is traversed, e.g. at a very low feed rate in the NC program, the axis is considered as moved for the complete path motion. However, it may not receive a new command position in every cycle due to quantisation. This means that the display “Axis is traversed” is not indicated in each cycle. Similarly, the axis is considered as traversed if override = 0 (traverse at velocity 0). By contrast, no velocity is commanded in the case of a FEEDHOLD, i.e. the axis is considered as not traversed.

Axis moves forwards	
Description	The last output setpoint would lead to a motion in the position direction of motion.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_IsMovingForward
Data type	BOOL
Value range	[TRUE = Axis traverses forwards, FALSE]
Access	PLC reads
Special features	This status flag is unable to detect whether the axis is actually in motion. Then use the status information pAC[axis_idx]^addr^.StateLR_Data.X_IsMoving.

Control unit enable, state	
Description	Indicates whether control unit is enabled for the axis.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_ControlLoopEnabled
Data type	BOOL
Value range	[TRUE = Control unit enabled for axis, FALSE = No control unit enable. Axis cannot be moved by the CNC.]
Access	PLC reads

Block number	
Description	Current block number of active NC block
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.D_BlockNr
Data type	DINT
Access	PLC reads

Axis supply	
Description	The command of the physical axes is executed by the logical axis specified in the NC channel.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.W_LinkToLogicalAxis
Data type	UINT
Value range	[= 0 – not linked, > 0 – logical axis number]
Access	PLC is reading

Cyclic drive values are valid	
Description	The actual values supplied by the drive to the cyclic process data are valid. This signal is reset if the control unit is active and the cyclic transfer of process data is interrupted.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_CyclicDriveDataOk
Data type	BOOL
Value range	[TRUE = Process data is valid, FALSE]
Access	PLC reads
Special features	This signal is dependent on the drive type as follows: SERCOS II: Signal is TRUE if the phase of the SERCOS ring is 4 and at least one of the bits 0x4000 or 0x8000 is set in the drive status word. SERCOS over EtherCAT: Signal is TRUE if the process datum WcState is 0 and at least one of the bits 0x4000 or 0x8000 is set in the drive status word. CANopen: Signal is TRUE if the process datum WcState is 0 and the drive status word contains a valid status according to the CANopen DS402 specification. PROFIBUS: Signal is TRUE if the life counter sign of the PROFIBUS slave is running. For all other drive types, the signal is set to TRUE immediately after CNC start-up.

Configured axis mode	
Description	The axis mode configured in the axis parameter list (P-AXIS-00015) is indicated.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.D_AxisMode
Data type	UDINT
Value range	See description of axis parameter P-AXIS-00015.
Access	PLC reads

Axis number of gantry master axis	
Description	If the axis is a gantry slave axis, the logical axis number of the master axis in this element (see P-AXIS-00070) is indicated; otherwise 0.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.W_GantryMasterNr
Data type	UINT
Access	PLC reads

Axis assignment	
Description	The display shows the physical axes to which the commanded values of the logical axes are of the channel output.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateIPO_Data.X_LinkToPhysicalAxis
Data type	UINT
Value range	[= 0 – not linked, > 0 – logical axis number]
Access	PLC is reading

Axis-specific interpolator, state	
Description	This element indicates various states of the axis-specific interpolator. The states are bit-encoded and may be active at the same time.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateIPO_Data.D_InternalIpoState
Data type	UDINT
Special features	The element is a bit string. Several states are therefore displayed at the same time.
Value range	The assignment of bits to states is listed in the table below.
Access	PLC reads

PLC constant	Value	Meaning
HLI_AX_INDP_INTERPOLATION	16#00000002	This is an independent axis for which set-points are generated.
HLI_AX_INDP_M_FUNC_PENDING	16#00000004	This is an independent axis which waits for acknowledgement of at least one technology function (M, H, S).
HLI_AX_INDP_ERROR	16#00000008	An error occurred in an independent axis.
HLI_AX_INDP_TIME_INTERPOLATION	16#00000010	Dwell time is active for an independent axis.
HLI_AX_MAN_MV_BACK_TO_START	16#00000020	Axis is moved because it received the command to reduce to 0 the position offset accumulated by motion in manual mode.
HLI_AX_MAN_MV_BACK_WAIT_STOP	16#00000040	The motion to reduce the position offset to 0 and which was generated by a motion of the axis in manual mode was interrupted and the axis is still in the deceleration phase to standstill.
HLI_AX_TRANSM_TO_PLC_IMPOSSIBLE	16#00040000	This is an independent axis where the axis-specific interface is assigned to output technology functions.
HLI_AX_FEEDHOLD	16#001000000	Axis-specific feedhold is active.

Type of axis coupling			
Description	This indicates whether the the axis is a slave axis in an axis coupling; see also [PROG// #AXIS LINK].		
Signal flow	CNC → PLC		
ST path	pAC[axis_idx]^ .addr^ .StatIPO_Data.W_AxLinkMode		
Data type	INT		
Value range	Value	PLC constant	Meaning
	-1	HLI_AXIS_LINK_NONE	No axis coupling active for this axis
	0	HLI_AXIS_LINK_NORMAL	The axis is a slave axis of an axis coupling
	1	HLI_AXIS_LINK_SPDL	The axis is a slave axis of a spindle coupling
	2	HLI_AXIS_LINK_GANTRY	The axis is a slave axis of a soft gantry combination
	3	HLI_AXIS_LINK_MIRROR	The axis is a slave axis of an axis coupling; the axis motion mirrors the master axis
Access	PLC reads		
Special features	The indicated axis coupling is only active when a master axis number is indicated at the same time.		

Axis number of master axis when axis coupling is active	
Description	When the axis is coupled to a master axis (see [PROG// #AXIS LINK]) the logical axis number of the master axis is indicated here.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^ .addr^ .StatIPO_Data.W_AxLinkMasterAxNr
Data type	UINT
Value range	0 : No coupling active > 0: logical axis number of the master axis
Access	PLC reads

Axis type (PCS)		
Description	This indicates the configured axis type (P-AXIS-00018).	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^ .addr^ .StateIPO_Data.W_Type	
Data type	UINT	
Value range	Value	Meaning
	0x0001	Linear axis
	0x0002	Rotary axis
	0x0004	Spindle
Access	PLC reads	
Special features	Is currently not programmed	

2.3.5.1 Drive of an axis

Drive type		
Description	Type of axis drive The drive type is specified for each axis in the entry kenngr.antr_typ.	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^ .addr^ .StateLR_Data.W_DriveType	
Data type	UINT	
Value range	Value	Meaning
	1	Conventional drive (+-10 V), not used
	2	SERCOS drive
	3	Profidrive
	4	Drive Simulation
	5	Lightbus
	6	Terminal axis (+-10V) via bus
	7	Real time Ethernet
	8	CANopen drive
	16	Virtual drive
	32	CAN drive (option)
Access	PLC reads	

Drive ready for power connection / drive torque enabled	
Description	The axis drives is ready for power connection / drive torque is enabled.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_ReadyPowerOn
Data type	BOOL
Value range	[TRUE = Ready for power connection, FALSE]
Access	PLC reads
Special features	SERCOS drives With SERCOS drives, this information is derived from bit 14 of the status word. To determine the drive state the datum ready_for_control_loop_on_r [► 37] must also be considered. CANopen drives With CANopen drives this signal is set when the drive is in the 'Operation enabled' state; in this state the drive is torque enabled. This drive state is indicated by the state word xxxx xxxx x01x 0111 transferred by the drive (binary notation, x = don't care).

Drive ready for operation	
Description	Drive control section and power supply are ready for operation.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_ReadyControlLoopOn
Data type	BOOL
Value range	[TRUE = Drive ready for operation, FALSE]
Access	PLC reads
Special features	SERCOS drives With SERCOS drives, this information is derived from bit 15 of the status word. In order to evaluate the drive state, the datum ready_for_power_on_r [▶ 36] (page 32 [▶ 35]) must also be considered. The various combinations are described below. If <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == FALSE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == FALSE)</pre> this means that the drive is not ready for power connection. If <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == TRUE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == FALSE)</pre> this means that the drive is ready for power connection and the internal drive initialisation procedures are completed. If <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == FALSE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == TRUE)</pre> this means that the drives control section and power supply are ready for operation but the drive has no torque. If <pre>(gpAx[axis_idx]^lr_state.ready_for_power_on_r == TRUE) AND (gpAx[axis_idx]^lr_state.ready_for_control_loop_on_r == TRUE)</pre> this means that the drive is ready for operation, i.e. “Feed rate enable, axis” [▶ 45] and “Drive on” [▶ 45] are set to TRUE and active. This means that the drive is torque enabled and follows the position command values in the NC kernel. CANopen drives This signal is set for CANopen drives when the drive is in valid state unequal to the ‘Not ready to switch on’. befindet.Der The ‘Not ready to switch on’ state is represented in the drive status word by the value xxxx xxxx x0xx 0000 (binary notation, x = don’t care). Drive has completed self-test and initialisation successfully.

Drive error	
Description	An error occurred in the drive. The drive is therefore interlocked.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Error
Data type	BOOL
Value range	[TRUE = Error occurred, FALSE]
Access	PLC reads
Special features	SERCOS drives With SERCOS drives, this information is derived from bit 13 of the status word. CANopen drives With CANopen drives this information is derived from bit 3 (bitmask 0x0008) of the state word.

Drive state		
Description	State information delivered by the drive via the fieldbus. This element contains the following data dependent on the drive type:	
	Drive type (P-AXIS-00020)	Datum
	SERCOS	Value of SERCOS parameter S-0-135 (drive state).
	CANopen	Value of CANopen object 0x6041 (drive state)
	PROFIDRIVE	Value of drive state word 1 (ZSW1)
	Lightbus	Value of drive signal 23 from the drive (drive state 'DriveState3')
	For further information refer to the drive documentation or the specification of the drive profile used.	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^^.addr^.StateLR_Data. W_NativeDriveState	
Data type	UINT	
Access	PLC reads	

Read drive data	
Description	Data transferred by the drive to the NC kernel. The content is application-specific This data is also provided in parallel in the state of the control unit, see read/write drive data cyclically [► 72]
Special features	Currently, data transfer can only be used for SERCOS drives. Therefore, enable transfer of the value in the axis parameter list to the drive, e.g.: <pre># Cyclically read the 4 byte PLC value # uns32_3 auf S-0-0819 antr_digital.typ.sercos.at[1].ident_nr 0819 antr_digital.typ.sercos.at[1].ident_len 4 antr_digital.typ.sercos.at[1].nc_ref LR_VAR3_IN</pre>
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_Word1 pAC[axis_idx]^^.addr^.StateLR_Data.D_Word2 pAC[axis_idx]^^.addr^.StateLR_Data.D_Word3 pAC[axis_idx]^^.addr^.StateLR_Data.D_Word4
Data type	UDINT
Access	PLC reads

DRIVE_STATE_MODE_0	
Description	present mode OF drive i.e. position loop control DRIVE_STATE_MODE_0 0x00000001
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode0
Data type	BOOL
Value range	
Access	PLC reads

DRIVE_STATE_MODE_1	
Description	DRIVE_STATE_MODE_1 0x00000002
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode1
Data type	BOOL
Value range	
Access	PLC reads

DRIVE_STATE_MODE_2	
Description	DRIVE_STATE_MODE_2 0x00000004
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode2
Data type	BOOL
Value range	
Access	PLC reads

DRIVE_STATE_MODE_3	
Description	DRIVE_STATE_MODE_3 0x00000004
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode3
Data type	BOOL
Value range	
Access	PLC reads
Special features	Is currently not programmed

DRIVE_STATE_MODE_4	
Description	DRIVE_STATE_MODE_4 0x00000005
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode4
Data type	BOOL
Value range	
Access	PLC reads
Special features	Is currently not programmed

DRIVE_STATE_MODE_5	
Description	DRIVE_STATE_MODE_5 0x00000016
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode5
Data type	BOOL
Value range	
Access	PLC reads
Special features	Is currently not programmed

DRIVE_STATE_MODE_6	
Description	DRIVE_STATE_MODE_6 0x00000007
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_Mode6
Data type	BOOL
Value range	
Access	PLC reads
Special features	Is currently not programmed

2.3.5.2 Compensation of an axis

Compensation of drive drifts	
Description	State of activation of drive drift compensation
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_DriftErrorComp
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads
Special features	Is currently not programmed

Compensation of spindle leadscrew error	
Description	State of activation of spindle leadscrew error compensation
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_PitchErrorComp
Data type	BOOL
Value range	[TRUE = Spindle leadscrew error compensation active, FALSE]
Access	PLC reads
Special features	Is currently not programmed

Compensation of temperature influence	
Description	Activation of drive temperature compensation
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_TempErrorComp
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads
Special features	Is currently not programmed

Compensation of backlash error	
Description	State of activation of compensation of drive backlash error
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_BacklashErrorComp
Data type	BOOL
Value range	[TRUE = Compensation of backlash error active, FALSE]
Access	PLC reads
Special features	Is currently not programmed

Feed forward	
Description	State of activation of feedforward control
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_FeedForward
Data type	BOOL
Value range	[TRUE = Feedforward control active, FALSE]
Access	PLC reads
Special features	Is currently not programmed

2.3.5.3 Measurement

Measuring probe state	
Description	State of measuring probe
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_MeasureEquipActive
Data type	BOOL
Value range	[TRUE = Measuring probe active, FALSE]
Access	PLC reads
Special features	Is currently not programmed

Measurement value valid	
Description	A valid measurement value was latched by the drive
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_MeasureValueOk
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads
Special features	Is currently not programmed

2.3.5.4 Link to ADS

Box identification	
Description	Device-dependent data defined when the system is configured (e.g. system manager) can be read out here.
Data type	HLITwincatBox
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.HLITwinCATBoxData
Access	PLC reads
Elements of the data type	
ST element	.AdsNetId
Description	This defines the AmsNetId of the TwinCAT computer to execute the function. An empty string can also be specified for the local computer. The PLC variable of this type is a string containing the AMS network identifier of the target device for which the ADS command is intended. The string consists of six numerical sections separated by dots. Each numerical section includes a number between 0 and 254. Valid AMS network address include: "1.1.1.2.7.1" or "200.5.7.170.1.7". If an empty string is transferred, the AMS network identifier of the local device is automatically accepted.
ST element	.W_AdsPort
Description	ADS devices in a TwinCAT network topology are identified by an AMS network address and a port number. The port number of the ADS device is specified by the system manager during configuration.
ST element	.W_AdsChannel
Description	

Device identification	
Description	The DeviceId specifies the I/O device where the function is to be executed. The device IDs are specified by the TwinCAT system manager during hardware configuration.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.HLITwinCATDeviceData
ST element	.D_Id
Data type	UDINT
Access	PLC reads
Special features	The device ID is currently used for ring global actions (e.g. phase switching or reading).

2.3.6 Axis control commands

Feed rate enable, axis	
Description	Axis-specific feed rate enable The feed rate enable must be set for all axes to be moved. If this is not the case, no path motion occurs.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	SERCOS drives With SERCOS drives, the datum is derived from bit 13 of the control word.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlBoolUnit_ReleaseFeedhold
Commanded, requested and return value	
ST element	.X_Command .X_Request
Data type	BOOL
Value range	[TRUE = Drive enable, Transition TRUE → FALSE: The drive is shut down in compliance with the acceleration parameters. FALSE = Drive STOP]
Return value	
ST element	.X_State
Data type	BOOL
Value range	[TRUE = Drive enable, FALSE = Drive STOP]
Redirection	
ST element	.X_Enable

Feedhold ON/OFF, axis	
Description	Axis-specific feedhold The axis-specific feedhold acts on an axis if the axis is not currently not in the path compound but in manual mode or is an independent axis. Otherwise, the global feedhold of the channel acts on the axis.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Data type	MCControlBoolUnit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. MCControlBoolUnit_Feedhold
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = feedhold on, FALSE = feedhold off]
Redirection	
ST element	.X_Enable

Feed override, axis	
Description	Axis-specific feed override The axis-specific feed override permits weighting the axis path velocity with an additional factor. The axis-specific feed override only acts on an axis if this axis is currently not in the path compound but is moved in manual mode or as an independent axis. Otherwise the global override of the channel acts on the axis. The axis-specific feed override also acts on single axes and spindles. With spindles the feed override also acts on the velocity acknowledgement of programmed M3/ M4 or MC_MoveVelocity commands. Velocity acknowledgement occurs when the weighted feed rate is reached and in extreme cases when the feedrate is 0. (see StateLR_Data.X_RevReached [► 79])
Data type	MCCControlUNS16Unit, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. MCCControlUNS16Unit_Override
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Unit	0,1 %
Data type	UINT
Value range	[0 ... P-AXIS-00109] The parameter P-AXIS-00109 is an axis-specific parameter. The typical value is 1000. See [AXIS].
Redirection	
ST element	.X_Enable

Feed override valid, axis	
Description	Axis-specific feed override valid.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. MCControlBoolUnit_OverrideValid
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Axis-specific feed override valid, FALSE]
Redirection	
ST element	.X_Enable

Drive ON	
Description	Drive ON
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	SERCOS drives With SERCOS drives, the datum is derived from bit 15 of the control word.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlBoolUnit_DriveOn
Commanded and requested values	
ST element	.X_Command .X_Request
Data type	BOOL
Value range	[TRUE = Drive ON, Transition TRUE → FALSE: The drive is shut down in the best possible manner. FALSE = Drive OFF]
Return value	
ST element	.X_State
Data type	BOOL
Value range	[TRUE = Drive ON, FALSE = Drive OFF]
Redirection	
ST element	.X_Enable

Control unit enable, command	
Description	Control unit enable ↔ axis-specific torque connection.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	SERCOS drives With SERCOS drives, the datum is derived from bit 14 of the control word.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControlBoolUnit_TorquePermission
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Torque connection, FALSE = Drive is torque-free]
Redirection	
ST element	.X_Enable

The diagram below shows the relationship between HLI control units and the SERCOS control word or the SERCOS status word for SERCOS drives.

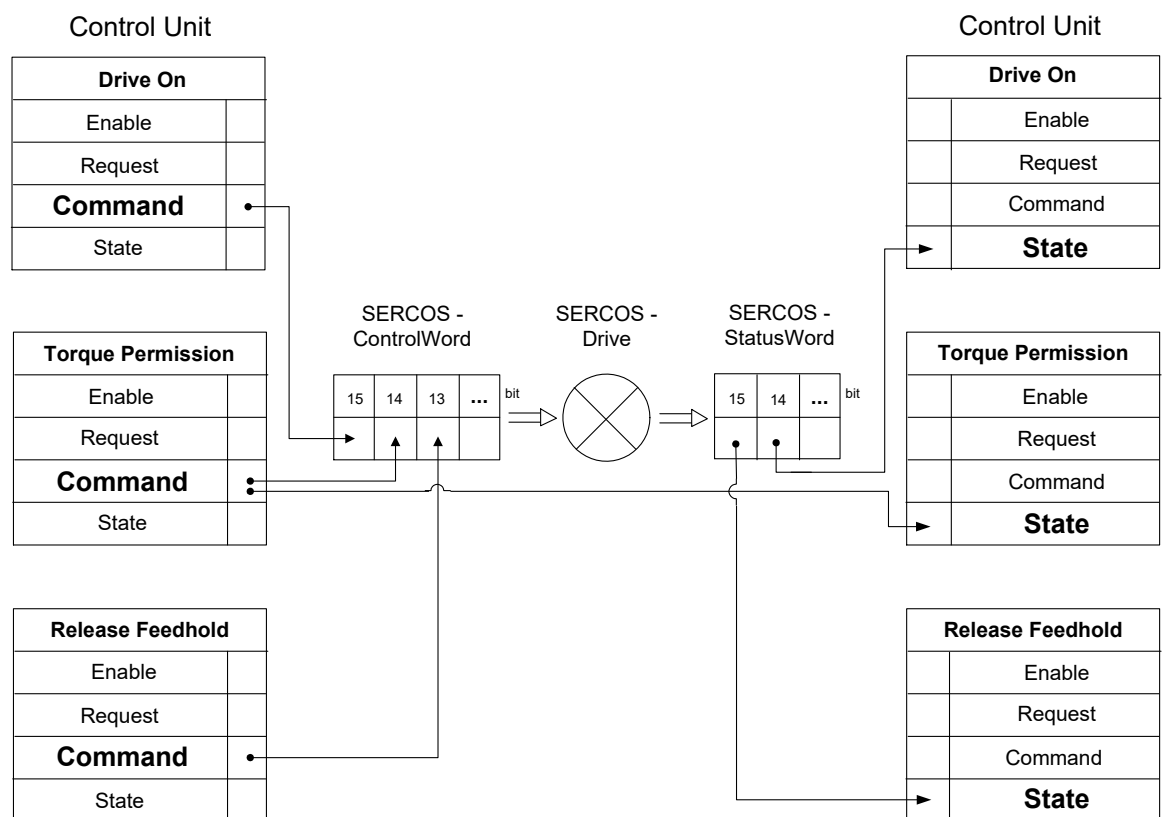


Fig. 10: Interaction of control units and SERCOS control or status words

Reference cam	
Description	Reference cam signal when homing.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	When this signal is commanded, it only acts if the parameter parameter <code>lr_hw[i].cam_direct_access = 0</code> is set in the axis machine data record of the associated axis. When the parameter parameter <code>lr_hw[i].cam_level</code> is used in the axis machine data record, the action of this command can be set from high-active to low-active. The default action is described below.
Access	PLC reads Request + State and writes Command + Enable
ST path	<code>pAC[axis_idx]^addr^.MCControlLr_Data.MCControlBoolUnit_ReferenceCam</code>
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reference cam switched, FALSE = Reference cam not switched]
Redirection	
ST element	.X_Enable

Reduced velocity, axis	
Description	When this signal is set, the axis velocity with G00 and G01 is reduced to the values defined in the axis parameters P-AXIS-00214 or P-AXIS-00155.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	If this axis is in a channel, the limits of the other axes participating in the motion are also considered. The effective value for reduced velocity is determined so that none of the axes participating in the motion overshoots their configured limit. The response is then identical to the command via a channel.
Access	PLC reads Request + State and writes Command + Enable
ST path	<code>pAC[axis_idx]^addr^.MCControlLrpo_Data.MCControlBoolUnit_ReducedFeed</code>
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity active, FALSE = Reduced velocity not active]
Redirection	
ST element	.X_Enable

Reduced velocity in zone 1, axis	
Description	By setting this signal, the axis velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 when the axis is located within the range defined by the parameters P-AXIS-00085 and P-AXIS-00093. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	If the axis is in a channel, the limits of the other axes participating in the motion and also located within their configured zone are considered.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.MCControllpo_Data.MCControlBoolUnit_ReducedFeedZone
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 1 active, FALSE = Reduced velocity in zone 1 not active]
Redirection	
ST element	.X_Enable

Reduced velocity in zone 2, axis	
Description	By setting this signal, the axis velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 when the axis is located within the range defined by the parameters P-AXIS-00097 and P-AXIS-00105. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	If the axis is in a channel, the limits of the other axes participating in the motion and also located within their configured zone are considered.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.MCControllpo_Data.MCControlBoolUnit_ReducedFeedZone2
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 2 active, FALSE = Reduced velocity in zone 2 not active]
Redirection	
ST element	.X_Enable

Suppressing Read-In Enable	
Description	Suppressing Read-In Enable When the read-in enable (setting NoEfg) is cleared, the interpolator does not read any new previously decoded NC motion information, i.e. the motion is stopped at the end of the current commands in the interpolator.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data. MCControlBoolUnit_NoEfg
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = No Read-In Enable, FALSE = Read-In Enable]
Redirection	
ST element	.X_Enable

Machining simulation, axis	
Description	Activates and deactivates an axis-specific machining simulation. During machining simulation, all axis-specific technology commands of the NC program are no longer output to the PLC but are acknowledged internally.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data. MCControlBoolUnit_MachiningSimu
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Machining simulation active, FALSE = Machining simulation inactive]
Redirection	
ST element	.X_Enable

Ignoring minimum tool velocity	
Description	If a value for minimum tool velocity is configured at tool change, the NC kernel monitors that this limit is not undershot by specifying an override. This control unit can switch this response off and the override acts on the axis as specified.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Control unit is only effective if the axis is a spindle.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. MCControlBoolUnit_IgnoreVbMinTool
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Undershooting minimum tool velocity permitted, FALSE]
Redirection	
ST element	.X_Enable

OTC offset	
Description	When this wear offset is set, wear in the direction of this axis can be compensated. When the SURF_NORM_ORI mode is used (wear in the direction of the surface normal), the offset value must be assigned in the third axis. Unit: 0.1µm
Data type	MCControlSGN32Unit, see description of control unit
Special features	The wear offset is distributed by the CNC over several cycles.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data. MCControlSGN32Unit_OTCOffset
Commanded, requested and return values	
ST element	.D_Command .D_Request .D_State
Data type	DINT
Value range	[-P-TOOL-00031, P-TOOL-00031]
Redirection	
ST element	.X_Enable
Move back manual mode offset	
Description	If manual mode is active in the channel and if the commanded axis fails to move, the axis is moved by this command so that afterwards manual mode offset is 0.
Data type	MCControlBoolUnit, see description of Control Unit
Special features	A rising edge (FALSE → TRUE) at X_Command starts the process. The signal is ignored if a manual mode motion is still active or manual mode offset is already 0.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data. MCControlBoolUnit_ManualMvBackToStart
Commanded, requested and return values	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	rising edge (FALSE → TRUE) triggers backward motion
Redirection	
ST element	.X_Enable

Stop the motion “Move back manual mode offset”	
Description	The motion that was started by the control unit command “Move back manual mode offset” is stopped by the control unit with this command.
Data type	MCControlBoolUnit, see description of Control Unit
Characteristics	A rising edge (FALSE → TRUE) initiates the command. Up to final axis standstill, the datum Axis-specific interpolator [► 33] shows that the stop process is active by the HLI_AX_MAN_MV_BACK_WAIT_STOP [► 33] bit.
Access	PLC reads Request + State and writes Command + Enable
ST Path	pAC[axis_idx]^^.addr^.McControlIpo_Data.MCControlBoolUnit_ManualMvBackStop
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Valuation	[TRUE = rising edge stops the motion, FALSE]
Redirection	
ST Element	.X_Enable

Tracking operation	
Description	The axis is set to tracking mode, i.e. the commanded setpoint is set identical to the actual value read in. Setting the setpoint and actual value equal is executed until X_Command = TRUE is.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	By setting the setpoint and actual value equal, the current position lag is = 0. This may result in an external force (weight of the axis) slowly changing the axis position (drift).
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControlBoolUnit_FollowUp
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = control loop open, FALSE]
Redirection	
ST element	.X_Enable

Running out gantry difference	
Description	If the axis is a gantry slave axis and both the master and slave are referenced, the gantry difference is run out.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlBoolUnit_GantryOn
Commanded, requested and return value	
ST element	. X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Gantry difference runout permitted, FALSE]
Redirection	
ST element	. X_Enable

Accepting reference position		
Description	Accept the reference position and mark the axis as referenced on a rising edge of this control unit. Dependent on the value of the parameter P-AXIS-00278 the actual axis position is set to the following value:	
	P-AXIS-00278	Reference position of the axis
	ABSOLUT	Value of P-AXIS-00152
	OFFSET	Encoder position of the drive + P-AXIS-00279
	PLC	Value of the ReferencePosition [► 58] in the control unit.
	PLC_OFFSET	Encoder position of the drive + value of the ReferencePosition [► 58] in the control unit.
	The X_State variable indicates whether the reference position was set manually and the coordinate system was shifted as a result. The manual setting can be cleared by CNC-controlled homing (G74). The reference position can also be set manually for an axis with absolute measuring system.	
Data type	MCControlDistCtrlUnit, see description of Control Unit	
Special features	Edge evaluation: The function is triggered on the rising edge at the command input.	
Access	PLC reads Request + State and writes Command + Enable	
ST path	pAC[axis_idx]^ .addr^ .McControlLr_Data.MCControlBoolUnit_SetReferencePosition	
Commanded, requested and return value		
ST element	.X_Command .X_Request .X_State	
Data type	BOOL	
Value range	[TRUE, FALSE]	
Redirection		
ST element	.X_Enable	

Reference position to be set		
Description	When the parameters P-AXIS-00278 have the value “PLC” or “PLC_OFFSET” in the axis parameter list, the value of the control unit (see also control unit SetReferencePosition [► 57]) is used to calculate the setting position when acceptance of the reference position is triggered via the HLI. In this case, there are options on how to use the value of this control unit. See below:	
	P-AXIS-00278	Reference position of the axis
	PLC	Value in this control unit.
	PLC_OFFSET	Encoder position of the drive + value in this control unit.
Data type	MCControlSGN32Unit, see description of Control Unit	
Access	PLC reads Request + State and writes Command + Enable	
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlSGN32Unit_ReferencePosition	
Commanded, requested and return value		
ST element	.D_Command .D_Request .D_State	
Data type	DINT	
Unit	0,1 µm bzw. 10 ⁻⁴ °	
Value range	[MIN_SGN32, MAX_SGN32]	
Redirection		
ST element	.X_Enable	

Clearing referencing	
Description	If an axis was homed by setting the reference position or by G74, this state can be cleared by using the actual control unit. If an axis has an absolute measuring system, the axis is subsequently considered as not referenced (it can be referenced again by a G74). This switches off the software limit switches temporarily, for example. Any offset activated by the control unit "Setting the reference position" is not cleared. The X_State variable indicates whether the axis is not currently referenced.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	As long as the control unit command (command_w) is TRUE, the axis is regarded as not referenced and can not be marked as referenced by triggering acceptance of the reference position (see also control unit SetReferencePosition [► 45]).
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlBoolUnit_ClearReferencePosition
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Measurement signal	
Description	This control unit transfers the measurement signal. When this control unit is used, set the entry kenngpr.probing_signal_via_plc to 1 in the parameter list of the appropriate axis.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Edge evaluation: Use the edge parameterised in the axis parameter list in the entry kenngpr.mess_neg_flanke to accept the measurement value. See also [AXIS].
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlBoolUnit_ProbingSignal
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Measurement value, axis	
Description	If the axis parameter P-AXIS-00257 or the NC command #MEAS [...SIGNAL=PLC] changed the measurement signal source to the control unit ProbingSignal [► 59], the measurement value can additionally be transferred via the HLI by activating this control unit.
Data type	MCControlSGN32Unit, see description of Control Unit
Special features	If this control unit is not activated, the axis actual position is used as the measurement value on the rising edge of X_Command of the ProbingSignal [► 59] control unit. The value assigned to the X_Command element of the control unit is transferred to the decoder without modification and is then available for further calculations.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlSGN32Unit_ProbingPosition
Commanded, requested and return values	
ST element	.D_Command .D_Request .D_State
Data type	DINT
Unit	0,1 µm bzw. 10 ⁻⁴ °
Value range	[MIN_SGN32 ... MAX_SGN32]
Redirection	
ST element	.X_Enable

Deactivate an axis (park)	
Description	Axis-specific deactivation of an axis. With a deactivated axis the following actions are not executed by the NC kernel: Error monitoring: errors signalled by the drive are not indicated by the CNC. HLL control bits are not transferred to the drive. A CNC reset does not trigger a drive reset. If you try to move a parked axis, the error message ID 70265 is output. No actions are executed in the drive.
Data type	MCControlBoolUnit, see description of Control Unit
Special features	PROFIDRIVE drives: When this control unit is active, the bit 0x80 in set in the control word 2 (STW2).
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControlBoolUnit_DeactivateAxis
Commanded, requested and return values	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = axis is deactivated, FALSE = axis is active (default mode)]
Redirection	
ST element	.X_Enable

2.3.6.1 Commanding axis couplings

Commanding axis couplings	
Description	This control unit defines axis couplings for each axis. The motion of the axis can be influenced additionally or exclusively by the motion of other axes. Further details are contained in the documentation [FCT-A9].
Data type	MCCControlDistCtrlUnit, see description of control unit
Access	PLC reads state and writes command + enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCCControlAxisCouplingUnit_AxisCoupling
Commanded values	
ST element	.AxisCouplingCommand
Signal flow	PLC → CNC
Data type	HLIAxisCouplingCommand [▶ 62]
Access	PLC writes
Return value	
ST element	.AxisCouplingState
Signal flow	CNC → PLC
Data type	HLIAxisCouplingState [▶ 62]
Access	PLC reads
Redirection	
ST element	.X_Enable

Axis couplings, state	
Description	Indicates whether the axis couplings are active for this axis and if so, which ones.
Signal flow	CNC → PLC
Data type	HLIAxisCouplingState
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data. MCControlAxisCouplingUnit_AxisCoupling. AxisCouplingState
Access	PLC reads
Elements of the data type	
Element	.desc[]
Data type	ARRAY [1..HLI_AxisCouplingMax] OF HLIAxisCouplingDesc [► 62]
Access	PLC reads
Element	.X_State
Data type	BOOL
Access	PLC reads

Axis couplings, command	
Description	This entry defines a coupling all coupling specification for the axis. The maximum number of definable coupling specifications is defined in the HLI_AxisCouplingMax constant. Further details on the definition of the coupling specifications are contained in the documentation [FCT-A9].
Signal flow	PLC → CNC
Data type	HLIAxisCouplingCommand
ST path	pAC[axis_idx]^ .addr^ .McControlLr_Data. MCControlAxisCouplingUnit_AxisCoupling. AxisCouplingCommand
Access	PLC writes
Elements of the data type	
Element	.desc[]
Data type	ARRAY [1..HLI_AxisCouplingMax] OF HLIAxisCouplingDesc [► 62]
Access	PLC writes
Element	.X_Semaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC accepts the commanded values when this element has the value TRUE. After acceptance, the CNC sets this value to FALSE. The PLC sets this element to TRUE when the commanded values are released for acceptance by the CNC. The commanded values can only be updated by the PLC if this element has the value FALSE.

Description of an axis coupling specification			
Description	This entry defines a coupling all coupling specification for the axis. The maximum number of definable coupling specifications is defined in the HLI_AxisCouplingMax constant. Further details on the definition of the coupling specifications are contained in the documentation [FCT-A9].		
Signal flow	PLC → CNC		
ST path	pAC[axis_idx]^addr^.McControlLr_Data. MCCControlAxisCouplingUnit_AxisCoupling.AxisCouplingCommand.desc[idx] pAC[axis_idx]^addr^.McControlLr_Data. MCCControlAxisCouplingUnit_AxisCoupling.AxisCouplingState.desc[idx]		
Data type	HLIAxisCouplingDesc		
Access	Command for axis coupling: PLC writes Status of axis coupling: PLC reads		
Elements of the data type			
ST element	.CouplingMode		
Data type	UINT		
Value range			
Value range	Constant	Value	Description
	HLI_AXIS_COUP- LING_INACTIVE	0	Coupling is not active.
	HLI_AXIS_COUP- LING_ZERO	1	Coupling factor is zero, used to deactivate an axis.
	HLI_AXIS_COUP- LING_DIRECT	2	Coupling factor is 1.
	HLI_AXIS_COUP- LING_MIRROR	3	Coupling factor is -1.
	HLI_AXIS_COUP- LING_FRACT	4	Coupling factor is a fraction defined by .desc[idx].FractNumerator / .desc[idx].FractDenominator.
ST element	. AxisNumber		
Data type	UINT		
Value range	[UINT_MIN, UINT_MAX]		
Description	Logical axis number of the source axis (the influencing axis). If the axis is to be moved by the NC program when the coupling is active, a coupling specification must be defined with the logical axis number of the axis and the coupling mode HLI_AXIS_COUPLING_DIRECT.		
ST element	.FractNumerator		
Data type	INT		

Value range	The permissible value range is [-32768, 32767]. A value of 0 in this element has the same effect as the coupling mode HLI_AXIS_COUPLING_INACTIVE.
Description	Numerator of coupling factor if coupling mode is specified as HLI_AXIS_COUPLING_FRACT. This element is not evaluated for all other coupling modes. The maximum permissible value for the coupling factor .desc[idx].FractNumerator /.desc[idx].FractDenominator is defined by the constant HLI_AXIS_COUPLING_FACT_MAX. If this value is exceeded, the error message P-ERR-70397 is issued.
ST element	.FractDenominator
Data type	INT
Value range	The permissible range is [-32768, 32767] excluding the 0. A value of 0 in this element results in output of the error message P-ERR-70396.
Description	Denominator of the coupling factor if the coupling mode is specified as HLI_AXIS_COUPLING_FRACT. This element is not evaluated for all other coupling modes. The maximum permissible value for the coupling factor .desc[idx].FractNumerator /.desc[idx].FractDenominator is defined by the constant HLI_AXIS_COUPLING_FACT_MAX. If this value is exceeded, the error message P-ERR-70397 is issued.

2.3.6.2 Distance control

Commanding distance control	
Description	This control unit can influence the distance control of the axis. This is dependent on whether it is selected in the axis parameters (see P-AXIS-00328).
Data type	MC_CONTROL_DISTANCE_CONTROL, see description Control Unit with usage check
Access	PLC reads state and writes command + X_enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data. MCControl_DistanceControl
Flow control of commanded values	
ST element	.X_CommandSemaphore
Signal flow	PLC → CNC
Data type	BOOL
Special features	Consumption data item

Access	CNC accepts the commanded values when this element has the value TRUE. After acceptance, the CNC sets this value to FALSE. The PLC sets this element to TRUE when the commanded values are released for acceptance by the CNC. The commanded values can only be updated by the PLC if this element has the value FALSE.
Commanded values	
ST element	.Command
Signal flow	PLC → CNC
Data type	HLIDistanceControlCommand [► 66]
Access	PLC writes
Distance control state	
ST element	.State
Signal flow	CNC → PLC
Data type	HLIDistanceControlState [► 66]
Access	PLC reads
Redirection	
ST element	.X_Enable

Command for distance control		
Description	This entry commands distance control.	
Signal flow	PLC → CNC	
Data type	HLIDistanceControlCommand	
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControl_DistanceControl. Command	
Access	PLC writes	
Elements of the data type		
ST element	.D_Transition	
Data type	UDINT	
Value range	Value	Constant
	0	HLI_DIST_CTRL_OFF
	1	HLI_DIST_CTRL_ON
	2	HLI_DIST_CTRL_FREEZE
	3	HLI_DIST_CTRL_REF
Description	See table: Transitions to command distance control [► 70]	
ST element	.D_Position	
Data type	DINT	
Value range	[DINT_MIN, DINT_MAX]	
Description	The meaning depends on the commanded transition: HLI_DIST_CTRL_ON: Sollposition der Werkstückoberfläche (SET_POS) HLI_DIST_CTRL_REF: Reference position of the workpiece surface (REF_POS)	

Distance control status		
Description	This entry read the distance control state.	
Signal flow	PLC → CNC	
Data type	HLIDistanceControlState	
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControl_DistanceControl. State	
Access	PLC reads	
Elements of the data type		
Element	.D_State	
Data type	UDINT	
Access	PLC reads	
Value range	Value	Constant
	0	HLI_DIST_CTRL_STATE_INACTIVE
	1	HLI_DIST_CTRL_STATE_ACTIVE
	2	HLI_DIST_CTRL_STATE_FREEZE
	3	HLI_DIST_CTRL_STATE_TURNING_OFF
	4	HLI_DIST_CTRL_STATE_ERROR
Description	See table: Distance control state [► 70]	
Element	.D_ActualPosition	
Data type	DINT	
Access	PLC reads	
Unit	0,1 µm bzw. 0,0001°	
Description	This datum indicates the current actual position of the workpiece surface detected by the sensing controller.	
Special features	This data item is only entered if distance control is activated in the axis parameters (see P-AXIS-00328).	
Element	.D_ActualOffset	
Data type	DINT	
Access	PLC reads	
Unit	0,1 µm bzw. 0,0001°	
Description	This data item indicates the current position offset of distance control by which the axis was moved due to deviations between the actual workpiece surface and the specified position (SET_POS). The following applies in stationary state (constant workpiece surface and position offset completely run out): Position offset = SET_POS – D_ActualPosition	

Special features	This data item is only entered if distance control is activated in the axis parameters (see P-AXIS-00328).
------------------	--

The tables below list the permissible values to command distance control and the defined states resulting.

Permissible transitions to command the distance control

Transition	Value	Meaning
HLI_DIST_CTRL_OFF	0	Distance control is deactivated. The program switches to the TURNING OFF state in which the position offset is run out. The state then switches automatically to the INACTIVE state.
HLI_DIST_CTRL_ON	1	Distance control is activated. When activated a command position for the workpiece surface must be transferred in the "position" datum. If no absolute encoder is used, distance control must first be referenced.
HLI_DIST_CTRL_FREEZE	2	The current position offset is frozen. Axis adjustment to the actual workpiece surface is ended.
HLI_DIST_CTRL_REF	3	Referencing distance control if no absolute encoder is used. Referencing is only permitted in the INACTIVE state. With this transition a reference position must additionally transferred in the "position" datum.
HLI_DIST_CTRL_ON_CONS T_DIST	4	Activating distance control with continuous distance specification. On activation, a set position must be specified. If the distance sensor supplies no absolute values, distance control must be referenced in advance.
HLI_DIST_CTRL_DRYRUN	5	Activate distance control for pure evaluation of data. No axis tracking in case of changes to the workpiece surface. When activated a command position for the workpiece surface must be transferred in the "position" datum. If no absolute encoder is used, distance control must first be referenced.
HLI_DIST_CTRL_CONST_DI ST	6	Activate distance control for pure evaluation of data. No axis tracking in case of changes to the workpiece surface. On activation, a set position must be specified. If the distance sensor supplies no absolute values, distance control must be referenced in advance.

Defined states of distance control

State	Value	Meaning
HLI_DIST_CTRL_STATE_INACTIVE	0	Distance control is deactivated. The specified offset ("actual_offset") is 0.
HLI_DIST_CTRL_STATE_ACTIVE	1	Distance control is active and adjusts the axis to the workpiece surface.
HLI_DIST_CTRL_STATE_FREEZE	2	Distance control is active. The offset ("actual_offset") is frozen, i.e. the axis is not adjusted to the workpiece surface.
HLI_DIST_CTRL_STATE_TURNING_OFF	3	Distance control was deactivated. The actual active offset ("actual_offset") is run out. As soon as the offset is 0, an automatic switch to INACTIVE state takes place.
HLI_DIST_CTRL_STATE_ACTIVE_CONST_DIST	4	Distance control is active and adjusts the axis to the workpiece surface. Continuous specification of the set distance of the tool to the workpiece surface.
HLI_DIST_CTRL_STATE_ERROR	5	Distance control is in error state, e.g. due to an incorrect state transition or due to an error in the position controller. Only a transition to TURNING OFF is possible to exit this state.
HLI_DIST_CTRL_STATE_DRYRUN_CONST_DIST	6	Distance control is active, however axis is not adjusted to workpiece surface. This allows the evaluation of data, for example filter effect without feedback from the control system. Continuous commanding of the set distance of the tool to the workpiece surface.
HLI_DIST_CTRL_STATE_DRYRUN_SETPOS	7	Distance control is active, however axis is not adjusted to workpiece surface. This allows the evaluation of data, for example filter effect without feedback from the control system.

2.3.7 Control commands of a drive

Reading/writing drive data cyclically	
Description	4 elements are provided for writing by the PLC in each interpolation cycle. They can be transferred to the drives via the subordinate drive protocol. The content and action are application-specific (dependent on the drives)
Data type	MCCControlUNS32Unit
Special features	Currently, data transfer can only be used for SERCOS drives. Therefore, enable transfer of the value in the axis parameter list to the drive, e.g.: <pre># write the 4 byte PLC value cyclically # uns32_1 auf S-0-0815 antr_digital.typ.sercos.mdt[1].ident_nr 0815 antr_digital.typ.sercos.mdt[1].ident_len 4 antr_digital.typ.sercos.mdt[1].nc_ref LR_VAR1_OUT # Cyclically read the 4 byte PLC value # uns32_3 auf S-0-0819 antr_digital.typ.sercos.at[1].ident_nr 0819 antr_digital.typ.sercos.at[1].ident_len 4 antr_digital.typ.sercos.at[1].nc_ref LR_VAR3_IN</pre>
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlLr_Data.MCControlUNS32Unit_D<i> where i = [1, 4]
Commanded and requested value	
ST element	.D_Command .D_Request
Data type	UDINT
Return value	
ST path	.D_State
Data type	UDINT
Special features	These values are also provided at the position controller interface. Siehe Antriebsdaten zyklisch lesen [► 35] (pAC[axis_idx]^addr^.StateLR_Data.D_Word1)
Redirection	
ST path	.X_Enable

Operation mode	
Description	Various drive operation modes can be commanded by the PLC (or operator): SERCOS drives: With SERCOS drives, this information is sent to bit 8 and bit 9 of the control word. This switches over between main and ancillary drive operation modes. Mode0 corresponds to the lowest operation mode control bit of the drive. Currently, variables Mode0 and Mode1 are used only for SERCOS drives. PROFIDRIVE drives: MCControlBoolUnit_Mode0 This control unit activates the drive function 'parking axis' by setting bit 7 in control word 2. The control unit state element indicates whether the function is active in the drive (value of drive status word 2 bit 7). Tracking mode is activated by the CNC internally for a parked axis. MCControlBoolUnit_Mode1: This control unit activates the drive function 'parking encoder' by setting bit 14 of the encoder control word. The control unit state element indicates whether the function is active in the drive (value of encoder status word but 14). Tracking mode is activated internally in the CNC for a parked encoder. MCControlBoolUnit_Mode2: Currently not in use.
Data type	MCControlBoolUnit
Special features	Elements MCControlBoolUnit_Mode3 ... MCControlBoolUnit_Mode6 are currently in use.
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlLr_Data.MCControlBoolUnit_Mode<i> where i = 0; 6]
Commanded, requested and return values	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Bit set, FALSE]
Redirection	
ST element	.X_Enable

2.3.8 Measuring with external measuring hardware

Interface for external measuring hardware	
Description	The CNC informs the PLC about the start or end of a measurement run via the external measuring interface so that the PLC can activate or deactivate an external measuring hardware accordingly. The measurement signal source must be set to PLC_EXT_LATCH_CTRL in order to activate this interface (see P-AXIS-00516 or [PROG//Extended programming])
Data type	LcControlExtLatchControl
Special features	This interface is only used to control the activation and deactivation of an external measuring hardware. The CNC signals detection of the measurement value or the detected measuring position via the control units ProbingSignal [► 45] or ProbingPosition [► 45].
Access	PLC writes Please + Done
ST path	pAC[axis_idx]^addr^ ExtLatchControl
Job	
ST element	.X_Please
Description	By setting X_Please the CNC signals to the PLC that the external measuring hardware is to be activated or deactivated since a measurement run is starting or was aborted.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	The CNC refreshes the data of the measuring interface only if this element is FALSE. After refreshment, the CNC sets this element to TRUE after the element X_Done was set to FALSE. The PLC reads the data of the measuring interface if this element has the value TRUE. After the data is transferred, the PLC sets the value to FALSE.
Parameter	
ST element	.ExtLatchOrder
Description	The CNC signals to the PLC in this datum the required measurement parameters such as activate/deactivate the measurement function or the number of the measurement input.
Data type	HLI_EXT_LATCH_ORDER [► 74]
Access	PLC reads
Acknowledgement	
ST element	.X_Done
Description	By setting the element X_Done to TRUE, the PLC signals to the CNC that the measuring command was executed.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	The PLC sets the value to TRUE if the measuring command was executed. The CNC sets the value to a FALSE before a new command.

Measuring command data	
Description	The CNC signals to the PLC in this parameter whether the measuring hardware must be activated or deactivated. In addition, the CNC informs the PLC of the measurement input to be used and the relevant measuring edge.
Data type	HLI_EXT_LATCH_ORDER
ST path	pAC[axis_idx]^addr^ExtLatchControl.ExtLatchOrder
Access	PLC reads
Measuring command identifier	
Description	ID of measuring command to be executed.
ST element	.Order_Id
Data type	UDINT
Value range	HLI_EXT_LATCH_ENABLE_PROBE: Start of measurement run, the measuring hardware must be activated. HLI_EXT_LATCH_DISABLE_PROBE: The measurement run was ended or aborted by reset. The measuring hardware must be deactivated again.
Access	PLC reads
Number of probing input	
Description	The CNC signals to the PLC in this datum the input to be used for measuring (see P-AXIS-00517)
ST element	.D_Input
Data type	DINT
Value range	Measuring input 1 – 255
Access	PLC reads
Relevant measuring edge	
Description	The CNC signals to the PLC in this datum the relevant measuring edge (raising/falling) to be used for measuring (see P-AXIS-00518)
ST element	.Edge
Data type	UDINT
Value range	HLI_MEAS_SIGNAL_LOW_ACTIVE: The measurement value is to be detected on the falling edge. HLI_MEAS_SIGNAL_HIGH_ACTIVE The measurement value is to be detected on the rising edge.
Access	PLC reads

The block diagram below shows an example of the signal profile of the control units ExtLatchControl [► 74], ProbingSignal [► 45] and ProbingPosition [► 45] during the measurement process with external hardware:

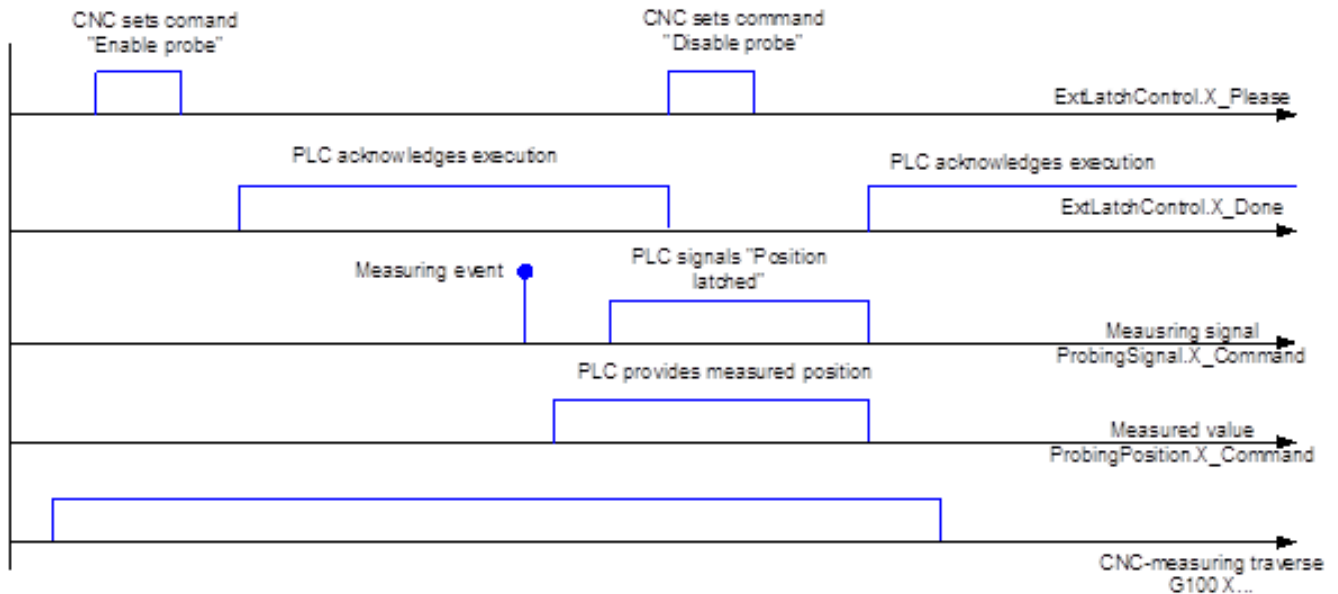


Fig. 11: Example signal profile of measurement with external hardware



Notice

When the external measuring interface is used, the probing signal of the control unit ProbingSignal [► 45] is not dependent on the relevant measuring edge P-AXIS-00518. A positive edge always signals the successful detection of a measurement value in the external measuring hardware.



Notice

If the control unit ProbingPosition [► 45] is not activated when the measuring event occurs, the measurement value of the current actual value at the time when the probing signal is used.

3 Spindle

3.1 Introduction

A spindle is an axis with extended properties. At the CNC end, a spindle is mapped by means of a separate motion controller (interpolator).

This means that the spindle axis can be moved not only by the NC program, motion jobs can also be generated at any time by the PLC.

3.2 Description of the spindle-specific interface

3.2.1 Speeds of a spindle

Nominal speed	
Description	Current speed setpoint of the spindle
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_ActiveRev
Unit	10 ⁻³ °/s
Data type	DINT
Access	PLC reads

Actual speed	
Description	Current actual speed of the spindle
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_CurrentRev
Unit	10 ⁻³ °/s
Data type	DINT
Access	PLC reads

Speed programmed	
Description	Speed setpoint of the spindle programmed via M03, M04 or S in the NC program
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.D_CmdRev
Unit	10 ⁻³ °/s
Data type	DINT
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

3.2.2 Positions of a spindle

Target position	
Description	Target position when positioning with M19
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.Statelpo_Data.HLIAXeDispData_Spindle.D_CmdPosition
Unit	10 ⁻⁴ °
Data type	DINT
Access	PLC reads

Actual position	
Description	Current actual position when positioning with M19
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.Statelpo_Data.HLIAXeDispData_Spindle.D_ActPosition
Unit	10 ⁻⁴ °
Data type	DINT
Access	PLC reads

3.2.3 Status information of a spindle

Operating state		
Description	Current operating state of the spindle	
Signal flow	CNC → PLC	
ST path	pAC[axis_idx]^^.addr^.Statelpo_Data.HLIAXeDispData_Spindle.D_Mode	
Data type	UDINT	
Value range	Value	Meaning
	1	M05 active
	2	M03 active
	4	M04 active
	8	M19 active
	16	Superimposed motion active (PLCopen)
	32	RPF active
Access	PLC reads	

Speed monitoring active	
Description	Speed monitoring is activated for the spindle.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_RevControlActiv
Data type	BOOL
Value range	[TRUE = Speed monitoring is active, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

Nominal speed reached	
Description	The spindle has reached the programmed speed setpoint.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_RevReached
Data type	BOOL
Value range	[TRUE = Speed setpoint reached, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

Spindle stopped	
Description	The spindle speed is lower than the value of vb_min_null (speed = 0) contained in the axis parameter list.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.StateLR_Data.X_RevZero
Data type	BOOL
Value range	[TRUE = Spindle stopped, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

Speed monitoring invalid

Description	If the speed of rotation of the spindle exceeds the limit speed of the position sensor of the spindle, speed monitoring is deactivated. The limit speed for the position measuring system is defined in the axis parameter list by parameter vb_regelgrenze.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateLR_Data.X_RevControlInvalid
Data type	BOOL
Value range	[TRUE = Speed monitoring invalid, FALSE]
Access	PLC reads
Special features	If the axis is not a spindle, the value is undefined.

Distance to go

Description	Distance to go when positioning with M19
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIpo_Data.HLIAXeDispData_Spindle.D_DistToGo
Unit	10 ⁻⁴ °
Data type	DINT
Access	PLC reads

Spindle command error

Description	A command sent to the spindle was not executed due to error.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.StateIpo_Data.X_spindle_order_error_r
Data type	BOOL
Value range	[TRUE = Command not executed, FALSE]
Access	PLC reads
Special features	Valid only in conjunction with external spindle command by the PLC

3.2.4 Control commands of a spindle

Spindle stop at program end	
Description	If this element is set to TRUE at end of the program, the spindle is stopped.
Data type	MCCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlBoolUnit_SpdIStopAtProgEnd
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Spindle is stopped at end of program, FALSE]
Redirection	
ST element	.X_Enable

Spindle reset	
Description	Commanding a reset for the spindle.
Data type	MCCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlBoolUnit_SpdIReset
Commanded and requested Value	
ST element	.X_Command .X_Request
Data type	BOOL
Value range	[TRUE Spindle reset commanded, FALSE]
Return value	
ST element	.X_State
Data type	BOOL
Value range	[TRUE = Spindle reset is executed, FALSE]
Redirection	
ST element	.X_Enable

Spindle error	
Description	Sets the spindle to error state
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. MCControlBoolUnit_SpdError
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Emergency stop, spindle	
Description	Commanding this control unit is only effective if the axis is a spindle. If this element is set to active for emergency stop (TRUE), a speed setpoint of zero is output. The axis is stopped at the emergency stop deceleration which is parameterised. This type of motion influence has maximum priority.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data. MCControlBoolUnit_EmergencyStop
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Emergency stop active, FALSE]
Redirection	
ST element	.X_Enable

3.2.5 External spindle command

3.2.5.1 Control unit of external spindle command

External spindle command	
Description	External spindle command. Commands, e.g. spindle stop or spindle positioning, can be issued to the spindle by this control unit over the HLI interface. Enter the other parameters of the command sent to the spindle in the structure HLI_EXT_TO_IPO_DATA.
Data type	MCCControlExtTolpo
Commanded values	
ST path	pAC[axis_idx]^addr^.McControllpo_Data.MCCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command
Signal flow	PLC → CNC
Data type	HLIExtTolpoData [► 83]
Access	PLC writes
Flow control commanded data	
ST path	pAC[axis_idx]^addr^.McControllpo_Data.MCCControlExtTolpo_ExtTolpo.X_Command-Semaphor
Signal flow	PLC → CNC
Data type	BOOL
Value range	[TRUE = Data is valid, FALSE = Data was accepted]
Special features	Consumption data item
Access	CNC accepts the commanded values when this element has the value TRUE. After acceptance, the CNC sets this value to FALSE. The PLC sets this element to TRUE when the commanded values are released for acceptance by the CNC. The commanded values can only be updated by the PLC if this element has the value FALSE.

3.2.5.2 User data for external spindle commands

The parameters for external spindle commands must be saved in the structure elements described below. Not all structure elements need to be completed dependent on the type of command for the spindle.

Programmed block feed	
Description	Programmed block feed
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.McControllpo_Data.MCCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command.D_VbProg
Unit	10 ⁻³ °/s
Data type	DINT
Access	PLC writes

G and M functions	
Description	Bit-encoded spindle parameters
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command. D_GeoGmFkt
Data type	UDINT
Value range	Only the following 2 bits are currently used: HLI_OPTIM_RICHTEN 0x00000010L Dress optimised rotary axis. When positioning the spindle, the target position is approached by the shortest possible path. HLI_ABSOLUT 0x00000100L Position information absolute
Access	PLC writes

Motion path	
Description	Motion path (relative or absolute) when positioning the spindle with M19.
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command. D_Fahrweg
Unit	10 ⁻⁴ °
Data type	DINT
Access	PLC writes

Type of spindle commands			
Description	The type of spindle commands is defined with this element.		
Signal flow	PLC → CNC		
ST path	pAC[axis_idx]^^.addr^.McControlIpo_Data.MCControlExtTolpo_ExtTolpo.HLIExtTolpoData_Command. W_SatzTyp		
Data type	UINT		
Value range	Value	Constant	Meaning
	1	HLI_NC_MOVE_LIN	Linear interpolation
	7	HLI_NC_MOVREF	Homing
	16	HLI_NC_MOVE_ENDLOS	Endless rotation
	20	HLI_NC_GETRIEBE	Spindle gear changes
	29	HLI_NC_SPINDEL_STOP	Stopping the spindle from endless rotation
	30	HLI_NC_SUPER_IMPOSED	accordingly PLCopen MC_SuperImposed
	31	HLI_NC_TABLE_SELECT	Selection of a table according to PLCopen MC_CamTableSelect
	32	HLI_NC_CAM_IN	accordingly PLCopen MC_CamIn
	33	HLI_NC_CAM_OUT	accordingly PLCopen MC_CamOut
	34	HLI_NC_GEAR_IN	accordingly PLCopen MC_GearIn
	35	HLI_NC_GEAR_OUT	accordingly PLCopen MC_GearOut
	36	HLI_NC_PHASING	accordingly PLCopen MC_Phasing
	38	HLI_NC_TOUCH_PROBE	accordingly PLCopen MC_TouchProbe
	39	HLI_NC_ABORT_TRIGGER	accordingly PLCopen MC_AbortTrigger
Access	PLC writes		

Direction of spindle rotation			
Description	Defining the direction of spindle rotation		
Signal flow	PLC → CNC		
ST path	pAC[axis_idx]^addr^.McControlIpo_Data.MCControlExtToIpo_ExtToIpo.HLIExtToIpoData_Command. W_DrehInfo		
Data type	UINT		
Value range	The following values apply when the spindle moves:		
	Value	Constant	Meaning
	0x0000	---	Negative direction of rotation
	0x0001	HLI_SPDL_POS_DREHR	Positive direction of rotation
Access	PLC writes		

4 Channel

4.1 Introduction

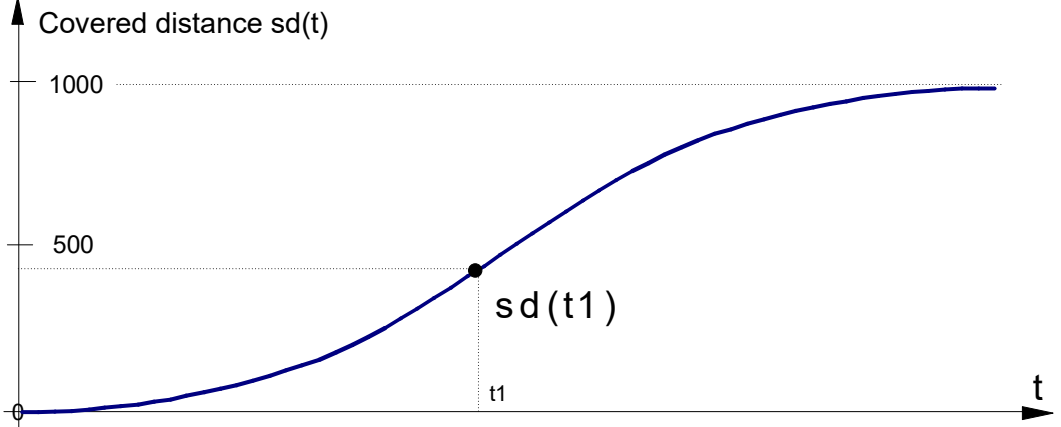
The command variables required for a machine motion are generated within a channel. In this case, input information which may be an instruction from an NC program, a manual motion block or an incremental motion command is converted into command positions for the machine drives. In order to generate a defined relative motion between tool and workpiece, command variables must be generated and coordinated for the machine axes involved. In this context, we refer to axes with a channel relationship. The number of axes controlled in a channel and the type of axes – translatory or rotary – are dependent on the machine concept.

For example, in the case of five-axis milling, three translatory and two rotary axes are controlled in one channel. Today, some machines offer the function of controlled their axes in several channels; an automatic multi-slide lathe is a typical example of this. The specific path motions on these machines are specified in the program for the individual channels and the corresponding command variables are generated independently of each other. However, it is possible to incorporate synchronisation points in each of NC programs to coordinate the channels.

Controls allow these machine concepts so that the generation of command variables is organised in separate channels. Accordingly, a separate NC program is started on each channel. Cross-channel synchronisation is executed either using NC commands or via the PLC.

4.2 Description of the channel-specific interface

4.2.1 Status information of a channel

Covered block motion path	
Description	Part of the path motion traversed in the current block in relation to the total path. This status datum contains the current block position referred to the path distance in space in the motion block in per mil $sd(t)$. 
Signal flow	CNC → PLC
Unit	0.1 %
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.D_CoveredDistance</code>
Data type	DINT
Access	PLC is reading
Special features	If a main axis participates in the motion, the covered path motion is in relation to the block path of the first three axes. If no main axis participates in the motion, the covered path motion is the position lag with the longest motion time in relation to the block path.

Currently covered path in the NC program (PCS)	
Description	Reads the currently covered path in the NC program since program start or since the last #DISTANCE PROG START CLEAR NC command. The calculation is based on the current position in the current NC block.
Signal flow	CNC → PLC
Unit	0.1 µm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_DistProgStartHigh pMC[channel_idx]^addr^.StateBahn_Data.D_DistProgStartLow
Data type	UDINT
Access	PLC is reading
Special features	In the NC this is an integer number which occupies 8 bytes in the memory. At the HLI the number is provided in the form of two 4-byte wide values. The value in D_DistProgStartLow contains the 4 lower bytes 0 to 3 and the value in D_DistProgStartHigh contains the higher bytes 4 to 7 of the 8-byte value present in the NC kernel. The read value can be used to command the block search to define the covered path in the NC program from where actual machining should effectively start.

Line counter, NC program	
Description	The datum indicates the NC program line which is the source of the command just processed by the interpolator. The value is derived from the number of NC program lines which the decoder has read since the NC program started. All the lines read the decoder are counted, i.e. repeatedly read lines, empty and comment lines. All commands to the interpolator resulting from decoding a NC program line are assigned to the associated line counter.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_BlockCount
Data type	UDINT
Access	PLC is reading

Programmed path feed	
Description	Path feed was programmed by the F<value> in the NC program.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_CommandFeed
Data type	DINT
Access	PLC reads

Path feed programmed taking into account real-time influences	
Description	Path feed which was programmed by the F<value> in the NC program weighted with the current real-time influences, e.g. override.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_CommandFeedActive
Data type	DINT
Access	PLC reads

Current path feed	
Description	Current path feed during interpolation.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_ActiveFeed
Data type	DINT
Access	PLC reads

Due to the architecture of the NC kernel in which various components operate asynchronously to each other, certain status information may be provided several times by the various control components.

For example, the signal X_ProgramEnd in the status flag of the decoder means that the decoder has completed decoding the program while the actual interpolation by the path interpolator may not yet be completed. The end of interpolation of the path axes is indicated by the signal X_ProgramEnd in the status data of the path interpolator.

End of program reached	
Description	Interpolator has reached end of program
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_ProgramEnd
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC reads

End of program reached	
Description	Decoder has reached end of program.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateDecoder_Data.X_ProgramEnd
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC reads
Special features	Caution: The machine cannot move any further since the interpolation signal is relevant for machine motion.

End of program reached	
Description	The look ahead function has reached the end of program
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBavo_Data.X_ProgramEnd
Data type	BOOL
Value range	[TRUE = End of program reached, FALSE]
Access	PLC reads
Special features	Required for diagnosis only.

Stop condition	
Description	Displays the condition why the current motion was stopped.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_StopConditions
Data type	DINT
Value range	See table: Value range of stop conditions [► 91]
Access	PLC is reading

Value range of stop conditions

Constant in PLC	Value	Explanation
SC_BIT_FEEDHOLD	0x0001	Path feed stop
SC_BIT_VFG	0x0002	No axis-specific feed enable.
SC_BIT_SINGLE_BLOCK	0x0004	Single step mode active.
SC_BIT_M00_OR_M01	0x0010	M00 (programmed stop), M01 (optional stop) is active.
SC_BIT_PLC_ACKNOWLEDGE	0x0020	Stop occurs due to waiting for an acknowledgement from the SPS. This may occur as a result of the output of M or H technology functions but is not restricted to them alone.
SC_BIT_OVERRIDE_ZERO	0x0040	Override = 0.
SC_BIT_DELAY_TIME	0x0200	Dwell time.
SC_BIT_CHANNEL_SYNC	0x0800	Channel synchronisation is active.
SC_BIT_IPO_INPUT_EMPTY	0x1000	Input FIFO of the interpolation is empty.
SC_BIT_IPO_INPUT_DISABLED	0x2000	Input of function blocks (e.g. motion blocks etc.) disabled.
SC_BIT_WAIT_FOR_AXES	0x8000	Stop occurs due to waiting until a commanded axis swap is completed.
SC_BIT_CHANNEL_ERROR	0x00010000	An error occurred in the channel.
SC_BIT_WAIT_TECHNO_ACK	0x00020000	Waiting for acknowledgement of M/H/ST technology functions.
SC_BIT_W_C_AFTER_COLLISION	0x00040000	After a detected collision, waiting for motion resumption.
SC_BIT_SLOPE_SUPPLY_PROBLEM	0x00080000	Block supply problem (only occurs in conjunction with HSC slope).
SC_BIT_BACK_INTERPOLATION	0x00100000	Back interpolation after tracking mode is active.
SC_BIT_BREAKPOINT_STOP	0x00400000	Stop because a breakpoint (stop point) is reached; available as of V2.11.2024.03, V2.11.2807.01, V3.1.3039.01.
SC_BIT_M0_STOP	0x02000000	Stop after an M00 function is reached
SC_BIT_M1_STOP	0x04000000	Stop after an M01 function is reached
SC_BIT_DEC_SYN_CHAN_EMPTY	0x10000000	Decoder is waiting for synchronisation. NC channel has no job.

Error occurred – awaiting clearance	
Description	An internal error has occurred. The interpolator waits for the error to be cleared.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitErrorRemoval
Data type	BOOL
Value range	[TRUE = Error occurred - interpolator waiting, FALSE]
Access	PLC reads

Error occurred– wait for external input	
Description	In syntax check mode, the decoder waits after an error for an external input (continue, abort)
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateDecoder_Data.X_WaitAfterError
Data type	BOOL
Value range	[TRUE = Error occurred – decoder waiting, FALSE]
Access	PLC reads

Interpolator active	
Description	Machine is to be/is moved.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_InterpolationActive
Data type	BOOL
Value range	[TRUE = Machine is to be/is moved, FALSE]
Access	PLC reads

Technology function acknowledgement	
Description	The interpolator waits for the acknowledgement of a technology function from the PLC.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitTechnoAcknowledge
Data type	BOOL
Value range	[TRUE = Waiting for acknowledgement of technology function, FALSE]
Access	PLC reads

Enable continuation of motion	
Description	The interpolator waits for an Enable to continue the motion after a stop in single-step mode.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitContinue
Data type	BOOL
Value range	[TRUE = Interpolator waiting motion to continue, FALSE]
Access	PLC reads

Dwell time active	
Description	The interpolator waits due to a programmed dwell time (G04).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_DwellTimeActive
Data type	BOOL
Value range	[TRUE = Interpolator waiting, FALSE]
Access	PLC reads

Axis group in position	
Description	All axes in the axis group have reached their programmed end positions.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_AxesInPosition
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads

Waiting for axis group in position	
Description	The interpolator waits in single-block mode or for all axes to be in position with programmed stop (M00) or optional stop (M01).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitAxesInPosition
Data type	BOOL
Value range	[TRUE, FALSE]
Access	PLC reads

Waiting for requested axis

Description	The look ahead function waits for a programmed axis request (see [PROG//#CALL AX]) that it contains the axis.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateBavo_Data.X_WaitForAxis
Data type	BOOL
Value range	[TRUE = Waiting for requested axis, FALSE]
Access	PLC reads

Block search active

Description	The interpolator works in block search mode.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateBahn_Data.X_BlockSearchActive
Data type	BOOL
Value range	[TRUE = active - Interpolator works in block search mode., FALSE]
Access	PLC is reading

Block search active

Description	Look ahead works in block search mode
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateBavo_Data.X_BlockSearchActive
Data type	BOOL
Value range	[TRUE = active – Look ahead works in block search mode, FALSE]
Access	PLC reads

Block search active

Description	The decoder works in block search mode
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^.addr^.StateDecoder_Data.X_BlockSearchActive
Data type	BOOL
Value range	[TRUE = active - Decoder works in block search mode, FALSE]
Access	PLC reads

Block search, distance from continuation position	
Description	If a NC program is started with block search mode, the NC program is processed in simulative mode (with no path motion) up to the specified continuation position. Block search is then in the HLI_BS_WAIT_FOR_PLC_OFF state and calculates the distance between the actual positions of the axis and the continuation position. If block search is in the HLI_BS_RETURNING_TO_CONTOUR state, this value is refreshed cyclically.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_BlockSearchPathDeviation
Unit	0.1 µm
Data type	UDINT
Value range	[0, MAX_SGN32]
Access	PLC is reading

Block search, state		
Description	Indicates the current state of the block search mode in the interpolator.	
Signal flow	CNC → PLC	
ST path	pMC[channel_idx]^addr^.StateBahn_Data.W_BlockSearchState	
Data type	INT	
Value range	Constant	Value
	HLI_BS_INACTIVE	0
	HLI_BS_WAIT_FOR_PLC_ON	1
	HLI_BS_ACTIVE	2
	HLI_BS_WAIT_FOR_PLC_OFF	3
	HLI_BS_WAIT_RETURN_TO_CONTOUR	4
	HLI_BS_RETURNING_TO_CONTOUR	5
	HLI_BS_WAIT_FOR_CONTINUE_CONTOUR	6
Access	PLC is reading	

Path velocity below limit	
Description	The path velocity undershoots the parametrised limit.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_SpeedLimitDetect
Data type	BOOL
Value range	[TRUE = active – Path velocity undershoots parametrised limit, FALSE]
Access	PLC reads

Rapid traverse velocity, axes move in the channel

Description	If the value is TRUE, the path axes move on the programmed path and the rapid traverse velocity was specified as motion velocity. TRUE is only indicated if at least one axis actually moves.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_RapidMode
Data type	BOOL
Value range	[TRUE = at least one path axis moves and rapid traverse velocity is specified, FALSE]
Access	PLC reads

Collision detected, wait to continue motion

Description	Indicates that after a collision is detected, the axis waits for a command to continue the motion.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_WaitContinueAfterCollision
Data type	BOOL
Value range	[TRUE = after a collision is detected, wait to continue the path, FALSE]
Access	PLC reads

Block number, current path motion

Description	If the N function [PROG//N function] is used in the active NC program to program NC block numbers, this datum indicates the NC block number of the NC block currently processed in the interpolator in this datum.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_BlockNumber
Data type	UDINT
Value range	[0, MAX_UN32]
Access	PLC reads

NC block, inserted	
Description	Indicates whether an additional NC block was inserted by the control unit during interpolation. Additional NC blocks may be created by functions such as polynomial contouring or tool radius compensation.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_BlockInserted
Data type	BOOL
Value range	[TRUE = the control unit inserted an NC block, FALSE]
Access	PLC reads

Manual mode is active, without parallel interpolation	
Description	Indicates whether exclusive manual mode is active. No interpolation is active at the same time.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_G200Active
Data type	BOOL
Value range	[TRUE = Manual mode is active without parallel interpolation, FALSE]
Access	PLC reads

Manual mode active, with parallel interpolation	
Description	Indicates whether superimposed manual mode is active. This means that the setpoints for the axes to be moved are calculated by superimposing the defaults for path interpolation and the mode interface of a particular axis.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_G201Active
Data type	BOOL
Value range	[TRUE = Manual mode is active with parallel interpolation, FALSE]
Access	PLC reads

Measurement process, active	
Description	Indicates that a measurement process was commanded by G100 or G108. The rising edge of the datum indicates that the measurement process was started. The value then remains TRUE until a measurement event was triggered for all the axes participating in the measurement run and for which measurement was activated.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_MeasureActive
Data type	BOOL
Value range	[TRUE = A measurement process is commanded but the measurement results have not yet been received for all measuring axes, FALSE]
Access	PLC reads

Delete distance to go, state	
Description	If a command is sent via the DeleteDistanceToGo [► 105] control unit, this value is TRUE as long as the NC block which exerts a linear motion to the target position of the next motion block is executed (short cut). It also remains TRUE if the active short cut is itself shortened by a new command. The signal is again reset when the active NC block is no longer related to the command of the DeleteDistanceToGo [► 105] control unit. See functional description [FCT-C28].
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.X_DeleteDistanceToGoActive
Data type	BOOL
Value range	[TRUE = linear motion is executed to the target position of the next motion block (short cut), FALSE]
Access	PLC reads

Time to next motion command containing G01, G02	
Description	Is rapid traverse velocity is specified for the current path motion, this datum indicates the time until the next motion block containing G01 or G02 is active. This time is only calculated and indicated if it is enabled in the start-up list in the parameter P-STUP-00070 by specifying FCT_LOOK_AHEAD_STANDARD FCT_CALC_TIME
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.D_TimeToNextFeedBlock
Data type	UDINT
Value range	[0, MAX_UN32]
Access	PLC reads

Distance to edge below limit

Description	This signals that the distance to the edge defined by the channel parameter P-CHAN-00222 (edge_machining.pre_dist) was undershot. As from this time, motion on the programmed path is effective at the path feed defined by the channel parameter P-CHAN-00223 (edge_machining.pre_feed).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.X_Signal_1
Data type	BOOL
Value range	[TRUE = active – Distance to edge undershoots parametrised limit, FALSE]
Access	PLC reads

Wait time at edge

Description	The signal indicates that the motion was stopped at the edge and the wait time specified by the channel parameter P-CHAN-00224 (edge_maching.wait_time) is expiring.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.X_Signal_2
Data type	BOOL
Value range	[TRUE = active – Wait time active, FALSE]
Access	PLC reads

Distance after edge below limit

Description	This signals that the tool moves away from the edge but the distance to the edge is still smaller than the value defined by the channel parameter P-CHAN-00225 (edge_machining.post_dist). The path feed valid for this section is defined by the channel parameter P-CHAN-00226 (edge_machining.post_feed).
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.X_Signal_3
Data type	BOOL
Value range	[TRUE = active - Distance after edge not reached, FALSE]
Access	PLC reads

Angle between active and next NC block	
Description	Indicates the angle between 2 consecutive NC blocks if both NC blocks are motion blocks containing programmed feed. The indicated value is in the range of [0, 1800000] which corresponds to [0°, 180°].
Signal flow	CNC → PLC
Unit	0.0001°
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIEdgeFunction_Data.D_AngleEnd
Data type	DINT
Value range	[0, 10000000]
Access	PLC reads
Special features	If the the next motion block is a rapid traverse block, the value value is 5000000 NC block leading to motion stop, the output value is 6000000 . In all other cases the default value 10000000 is indicated.

Velocity at end of current NC block	
Description	Indicates the velocity at transition from currently interpolated to next motion block which results from evaluating the geometrical situation and the velocity of the next block. If the dwell time is programmed or a predictable motion stop is programmed at the transition between the NC blocks caused by the output of a technology function with corresponding synchronisation start in the next block, the value 0 is indicated.
Signal flow	CNC → PLC
Unit	1 µm/s
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBlockDynamic_Data.D_VelEndGeo
Data type	DINT
Value range	[0, MAX_SGN32]
Access	PLC reads

4.2.1.1

Status information on tool orientation

The 3 vectors **tb_vec** (path tangent), **tn_vec** and **fn_vec** form a complete right-handed space coordinate system (moving trihedron).

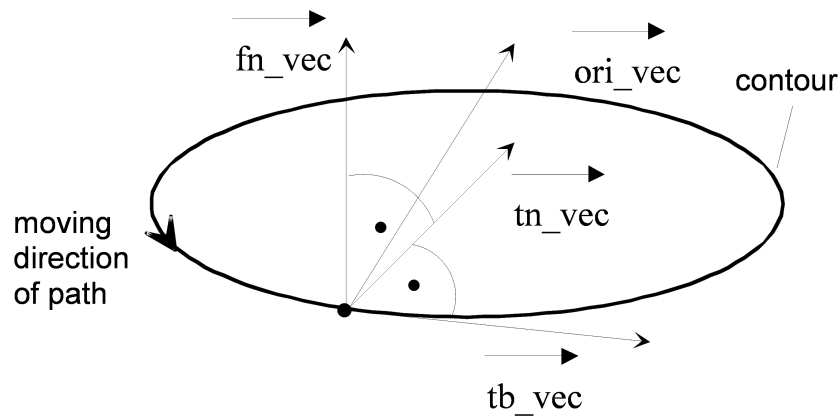


Fig. 12: Vectors of the tool coordinate system

NOTE:

When tool geometry compensation is active and when machining with the face of the tool, the result is the tool direction vector **ori_vec** obtained from the face normal vector **fn_vec**, the path tangent vector **tb_vec** and the advance and lateral angle.

(END)

The vectors of the tool coordinate system are integrated in the data structure **HLIToolPathDisp-Data**. The tables below contain the description of the elements for this data structure:

Tool direction vector	
Description	Components of the tool direction vector or recalculation from face normal vector and path tangent vector (see Figure)
ST path	<code>pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.ori_vec[vec_idx]</code>
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length 10^6 .

Path tangent vector	
Description	Components of the path tangent vector (see Figure).
Unit	Direction vector normalised to length 10^6
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.tb_vec[vec_idx]
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length 10^6 . The last valid motion direction is retained.

Normal vector to the path tangent	
Description	Components of the resulting vector, cross product of ori_vec $\times$ tb_vec, or cross product of fn_vec $\times$ tb_vec (see Figure).
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.tn_vec[vec_idx]
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length 10^6 .

Auxiliary vector of the complete trihedron, face normal vector	
Description	Components of the resulting vector, cross product of tb_vec $\times$ tn_vec, or face normal vector (see Figure).
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIToolPath_Data.fn_vec[vec_idx]
Data type	ARRAY [1..HLI_CS_AXES_NR] OF DINT
Special features	The direction vector is normalised to length 10^6 .

4.2.1.2 Diagnosis upload

Diagnosis upload	
Description	While the CNC is running, the PLC can command an upload of diagnosis data using this control unit. The control unit is enabled by X_Enable = TRUE.
Data type	MCControlBoolUnit, see description of Control Unit
Data type	MCControlBoolUnit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^^.addr^.MCControlBahn_Data. MCControlBoolUnit_DiagnosisUpload
Commanded, requested and return values	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = diagnosis upload activated, FALSE = diagnosis upload off]
Redirection	
ST element	.X_Enable
Special feature	Note: The data item X_Command must remain at TRUE until X_State reverts to FALSE. Otherwise, the data is not complete since the diagnosis data upload is aborted.

4.2.2 Control commands of a channel

Skip mode, NC block	
Description	Activates/deactivates skip mode at interpreter level for the NC program. The status of skip mode is only evaluated at the start of the NC program. Switchover during execution of an NC program has no effect.
Data type	MCControlBoolUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlDecoder_Data.MCControlBoolUnit_ProgramBlockIgnore
Commanded, requested and return values	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Skip mode NC block ON, FALSE = Skip mode NC block OFF, FALSE]
Redirection	
ST element	.X_Enable

Single-block mode at interpreter level NC block	
Description	Activates/deactivates single-block mode at interpreter level. A restart must be present for each block
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Is currently not supported
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlDecoder_Data.MCControlBoolUnit_SingleBlock
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = single-block mode ON, FALSE]
Redirection	
ST element	.X_Enable

Enable processing of next NC block	
Description	Enable to process the next NC block. This element serves to synchronise the geometry chain with the PLC. For this purpose, the PLC blocks the single-block step enable and thus stops the interpreter.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Is currently not supported.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlDecoder_Data.MCControlBoolUnit_ContinueMa- chining
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Continuation NC program decoding	
Description	Continues decoding an NC program after an error has occurred if syntax check (simulation mode) and interactive step enabling of decoding is activated (decoder parameter, characteristic parameter: syn_chk.interaktiv = 1).
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Is currently not supported.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlDecoder_Data.MCControlBoolUnit_ReleaseStop
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Stop NC program decoding	
Description	Stops decoding an NC program if interactive enabling of further processing of the NC program was activated in Syntax Check Simulation mode.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Is currently not supported.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlDecoder_Data.MCControlBoolUnit_DecStop
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Channel mode			
Description	Selection of a special channel mode such as syntax check or machining time calculation		
Data type	MCControlSGN32Unit, see description of Control Unit		
Access	PLC reads Request + State and writes Command + Enable		
ST Path	pMC[channel_idx]^ .addr^ .MCControlDecoder_Data. MCControlSGN32Unit_Execution-Mode		
Commanded, requested and return values			
ST Element	.D_Command .D_Request .D_State		
Data type	DINT		
Value range	Value	Constant	Meaning
	0x0000	ISG_STANDARD	Normal mode
	0x0001	SOLLKON	Block search
	0x0002	SOLLKON	Nominal contour visualisation simulation with output of visualisation data
	0x0802	SOLLKON_SUPRESS_O UTPUT & SOLLKON	Nominal contour visualisation simulation without output of visualisation data
	0x0004	ON_LINE	Online visualisation simulation
	0x0008	SYNCHK	Syntax check simulation
	0x0010	PROD_TIME	Simulation machining time calculation (No function with TwinCAT)
	0x0020	ONLINE_PROD_TIME	Simulation online machining time calculation
	0x0040	MACHINE_LOCK	Dry run without axis motion
	0x0080	ADD_MDI_BLOCK	Extended manual block mode: the end of a manual block is not evaluated as a program end. It permits the commanding of further manual blocks.
	0x0100	KIN_TRAFO_OFF	Overwrites automatic enable for kinematic transformations by a characteristic parameter defined in the channel parameters (sda_mds*.lis).
	0x1000	BEARB_MODE_SCENE	When SCENE mode is enabled, the output of #SCENE commands is activated on the interface (see also [FCT-C17// Scene contour visualisation]). An additional client is linked to this output via DataFactory / CORBA.
	0x2000	SUP- PRESS_TECHNO_OUT- PUT	Without output of technology functions (M/H/T). Set implicitly in connection with syntax check

Redirection	
ST element	.X_Enable

Index of position offset group	
Description	A position offset defines an additional offset. For instance, this takes into consideration different clamping positions of a workpiece in a machine workspace. Position offsets are defined in the position offset data. The position offsets for each axis are specified in a position offset group. A position offset group is selected by the group index. Position offset data is evaluated by the CNC at program start.
Data type	MCCControlSGN16Unit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlDecoder_Data. MCCControlSGN16Unit_ClampPosition
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Data type	INT
Value range	[0, 67]
Redirection	
ST element	.X_Enable

Feedhold ON/OFF	
Description	Channel-specific feedhold. When this element is set to TRUE , the feed velocity is immediately ramped down to feed = 0 during interpolation based on the acceleration settings.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	The channel-specific parameter P-CHAN-00097 can be used to exert additional influence on the parameterised accelerations to be used. Feedhold ON/OFF also be commanded by the control unit for switching operation modes (see Sec. 8.2). A HOLD command results in stopping the channel; a RESUME command cancels the stop command. If the PLC is registered at both control units, pay attention to the following safety note:
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_FeedHold
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Feedhold ON, FALSE]
Redirection	
ST element	.X_Enable



⚠ CAUTION

Stop command to CNC is not executed.

Possible damage to machine.

Similar to other control units, a command from an external application, e.g. an HMI, is displayed in **X_Request** . If the PLC operates this control unit and therefore sets **X_Enable** to TRUE, the command from the external application only becomes effective if **X_Request** to **X_Command** . However, the semaphores must be operated as usual.

This must also be considered if the PLC operates the control unit in order to switch operation modes (see sec. 8.2) and therefore sets **X_McmEnable** to TRUE. A HOLD command sent by this control unit triggers the NC kernel to send a request to activate feedhold. This is then displayed in the **X_Request** of the control unit described here and only becomes effective when the PLC copies **X_Request** to **X_Command** . This also applies to cancelling feedhold.

The figures below depict these functions.

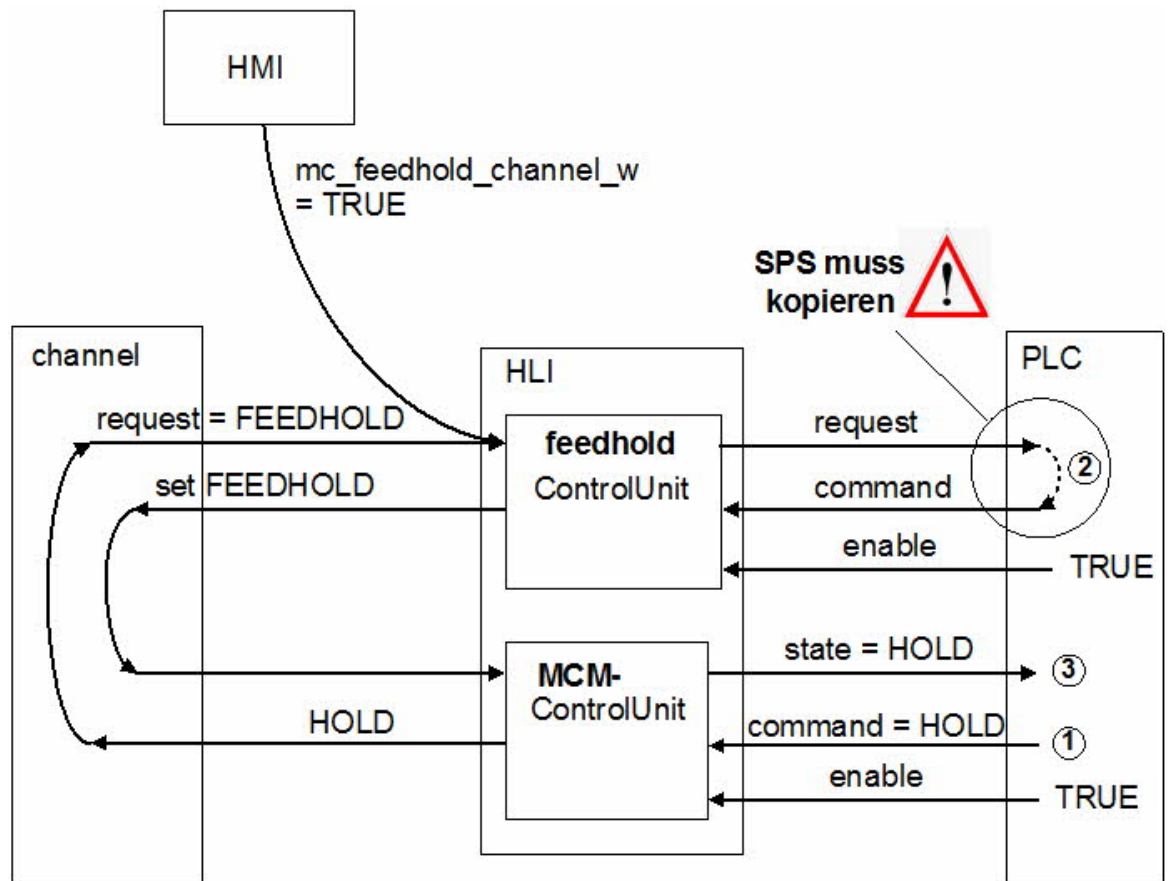


Fig. 13: Interaction between feedhold and NC channel stop

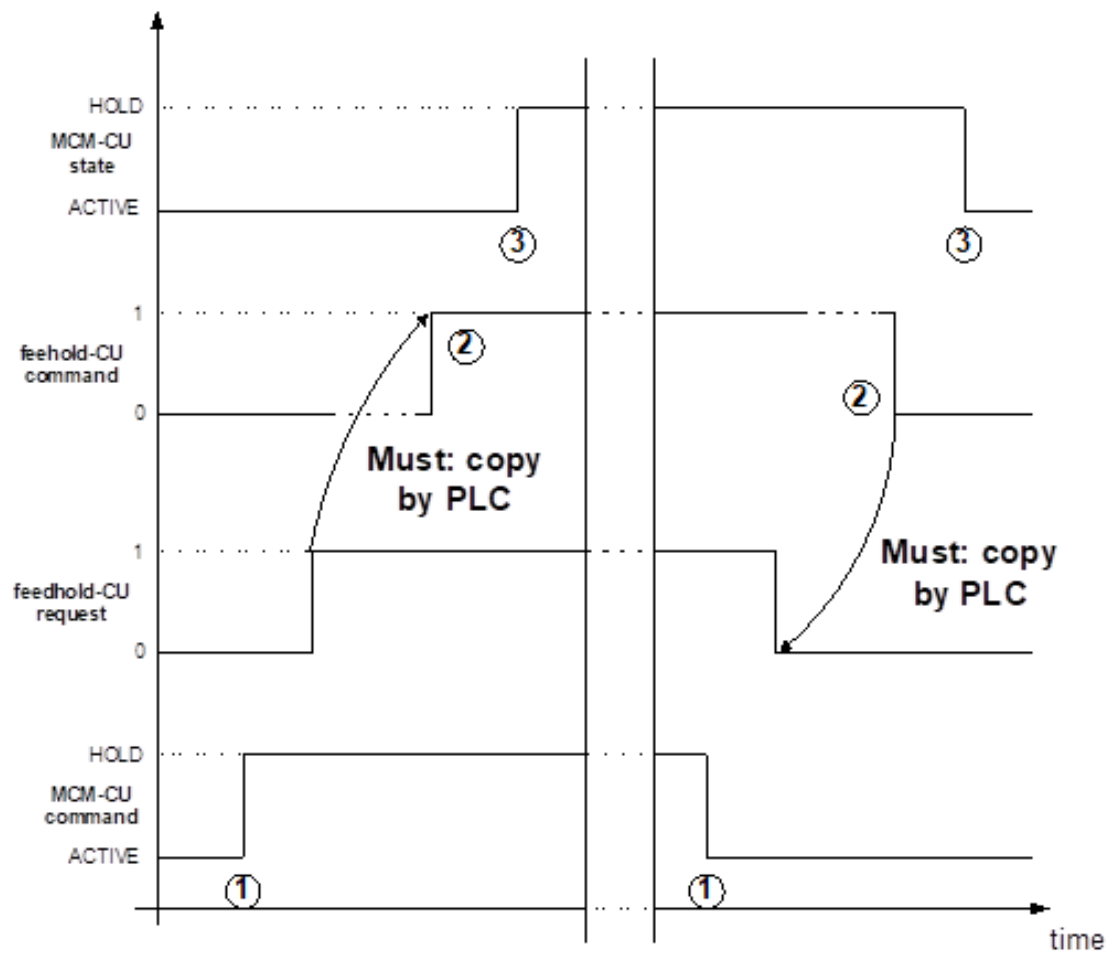


Fig. 14: Time sequence of feedhold and NC channel stop

Emergency stop, channel	
Description	If this emergency stop element is set active (TRUE), interpolation is aborted immediately by output of the zero setpoint or by a deceleration as specified in the emergency deceleration. The NC kernel reverts to error state. This type of motion influence has maximum priority.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_EmergencyStop
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Emergency stop active, FALSE]
Redirection	
ST element	.X_Enable

Single-block mode	
Description	Activating/deactivating single-block mode. Single-block mode refers only to motion blocks. As long as single-block mode is activated, the system is decelerated to feed rate = 0 at the end of each motion block. The following blocks can then only be executed by setting the element "continue motion" if all axes are located in the control window.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_SingleBlock
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Optional stop	
Description	Activating/deactivating optional stop. If the function M01 (optional stop) is programmed in the current block of the NC program, set this element to the value TRUE to stop at block end (ramped-down deceleration complying with the permissible accelerations). The following block can be enabled by activating the element "continue machining" if the NC kernel indicates that all axes are in the control window by resetting status flag X_WaitAxesInPositiont.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlBahn_Data.MCControlBoolUnit_M01StopEnable
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Optional stop active, FALSE]
Redirection	
ST element	.X_Enable

Resumption of motion	
Description	If program execution is interrupted by selecting "Single block mode" [► 105] or "Optional stop" [► 105] or by M00, this control unit can resume NC program execution. A falling edge for the command value (command_w) of the control unit "Continue motion", i.e. a transition from TRUE to FALSE, leads to a resumption of NC program execution. The condition for this is that all axes are located in the control window.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	Falling edge of the command resumes NC program execution.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCControlBahn_Data.MCControlBoolUnit_ContinueMotion
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Machining simulation ON/OFF	
Description	Activating/deactivating machining simulation. During machining simulation, all technology functions of the NC program are not output to the PLC but are acknowledged internally.
Data type	MCCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlBoolUnit_MachiningSimulation
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Machining simulation active, FALSE = Machining simulation inactive]
Redirection	
ST element	.X_Enable

Feed override	
Description	Feed override allows the programmed path velocity to be weighted with an additional factor
Data type	MCCControlUNS16Unit, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlUNS16Unit_OverrideFeedrate
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Unit	0,1 %
Data type	UINT
Value range	[0, P-CHAN-00056] The parameter P-CHAN-00056 is a channel-specific parameter. The typical value is 1000. See [CHAN].
Redirection	
ST element	.X_Enable

Rapid traversing override	
Description	The rapid traverse override can weight G0 path motions by an additional factor. See also special features.
Data type	MCCControlUNS16Unit, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^ .addr^ .MCCControlBahn_Data.MCCControlUNS16Unit_OverrideRapid-Move
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Unit	0,1 %
Data type	UINT
Value range	[0, 1000]
Redirection	
ST element	.X_Enable
Special features	
Parameterisation/ mode of operation	Rapid traverse override is only active if this function is also activated in the channel parameter list. Otherwise, there is no distinction made between feed and rapid traverse blocks. Setting options in the channel parameter P-CHAN-00181: Rapid traversing override is inactive. Rapid traversing override acts on feed and rapid traversing blocks. Rapid traversing override is active. Feed override acts on feed blocks and the minimum of feed and rapid traversing override acts on rapid traversing blocks. Rapid traversing override is active. Feed override acts on feed blocks and only rapid traversing override acts on rapid traversing blocks.

Interrupt output of command values to real axes	
Description	This interrupts the output of NC channel command values to physical axes. The NC channel is stopped and assignment to the real axes is disabled. Physical axes can then be requested and moved by another channel. A different logical axis can be linked to a physical axis here. After clearing this interruption, the axes can be requested again and the original channel continues its motion.
Data type	MCControlBoolUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_SuspendAxisOutput
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

External path velocity specified	
Description	External path velocity specified. The set path velocity is activated by the control unit MC-ControlBoolUnit_ExtCommandSpeedValid.
Data type	MCCControlUNS32Unit, see description of control unit
Special features	The path velocity transferred by this interface is automatically restricted to the limits defined in the axis parameters.
Unit	1 µm/s
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlUNS32Unit_ExtCommand-Speed
Commanded and requested Value	
ST element	.D_Command .D_Request
Data type	UDINT
Return value	
ST element	.D_State
Data type	UDINT
Special features	The D-State indicates the actual path velocity used in the interpolate, including any influence by override. By default the externally specified velocity only acts on machining motions (G01, G02, G03). The channel parameter P-CHAN-00102 (plc_command_rapid_feed) sets that the externally specified velocity also acts on rapid traverse motions (G00).
Redirection	
ST element	.X_Enable

Activation of external path velocity	
Description	Activating the velocity commanded in the control unit MCControlUNS32Unit_ExtCommandSpeed. To reach the commanded velocity, all axes involved in the motion are accelerated or decelerated.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ExtCommandSpeedValid
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable

Input disable, interpolator	
Description	Disabling input of blocks in the interpolator. When the control unit is enabled, the interpolator stops after it has processed the blocks already input. The channel parameter P-CHAN-00267 can define the event when an activated input disable becomes effective, e.g. effective as of the next rapid traverse block.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	See safety not below.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_InputDisable
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST element	.X_Enable



⚠ CAUTION

Stop command to CNC not executed.

Possible damage to machine.

Similar to other control units, a command from an external application, e.g. an HMI, is displayed in **X_Request**. If the PLC operates this control unit and therefore sets **X_Enable** to TRUE, the command from the external application only becomes effective if **X_Request** to **X_Command**. However, the semaphores must be operated as usual.

This must also be considered if the PLC operates the control unit in order to switch operation modes (see sec. 8.2) and therefore sets **X_McmEnable** to TRUE. A HOLD command given by this control unit triggers the NC kernel to request input disable. This is then displayed in the **X_Request** of the control unit described here and only becomes effective when the PLC copies **X_Request** to **X_Command**. The same also applies to cancelling input disable.

The figures below depict these functions.

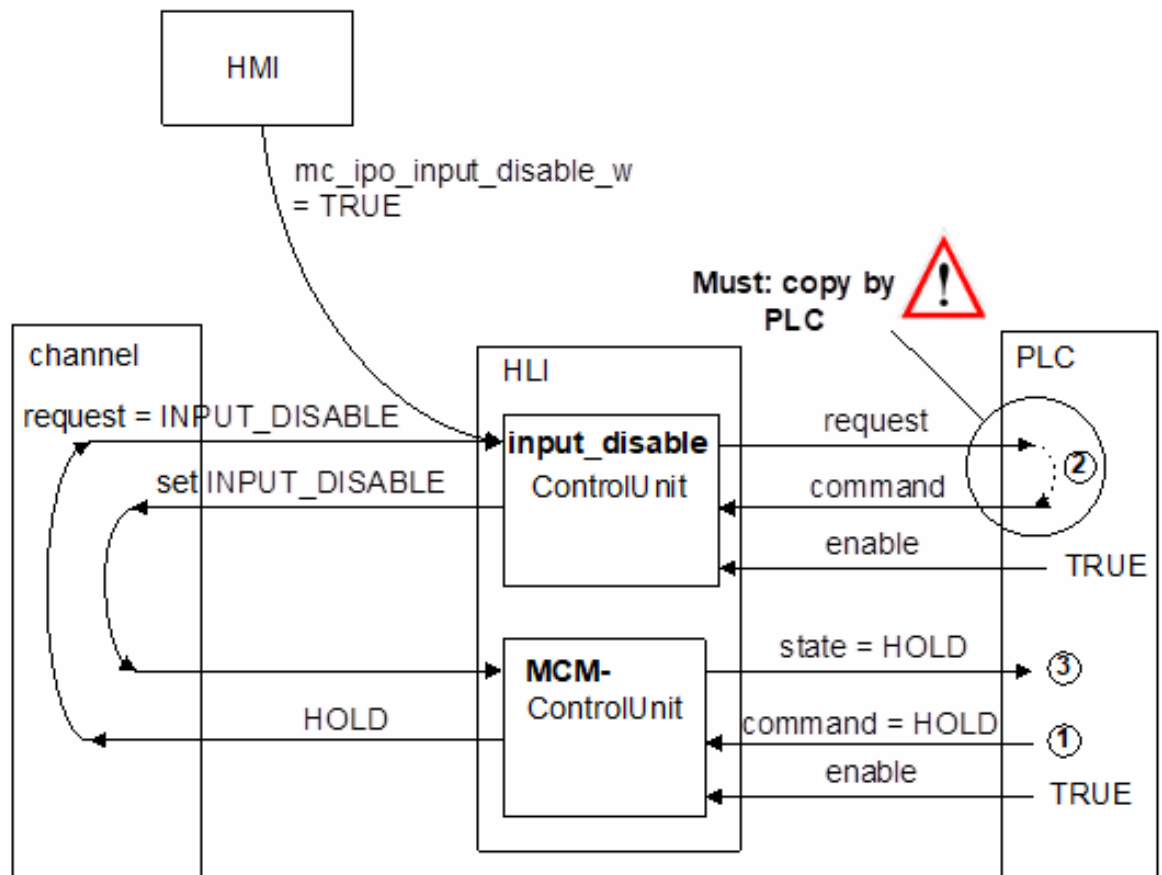


Fig. 15: Interaction of input disable and NC channel stop

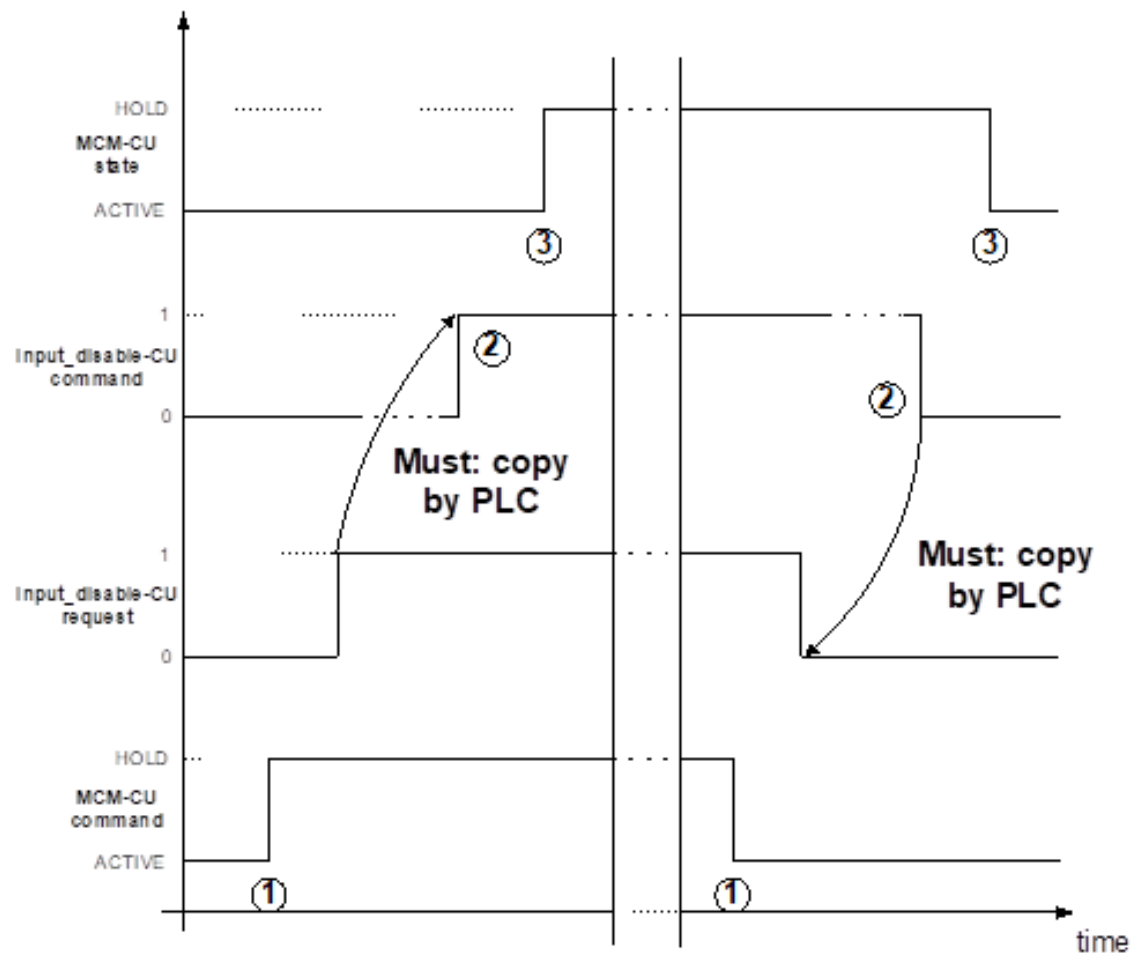


Fig. 16: Time sequence of input disable and NC channel stop

Reduced velocity, channel	
Description	When this signal is set, the path velocity is reduced to the values defined in the axis parameters P-AXIS-00214 and P-AXIS-00155.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Special features	The limits of the axes participating in the motion are taken into consideration. The effective value for reduced velocity is determined so that none of the axes participating in the motion overshoots its configured limit.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ReducedFeed
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity active, FALSE = Reduced velocity not active]
Redirection	
ST element	.X_Enable

Reduced velocity in zone 1, channel	
Description	When this signal is set, the path velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 if the axis is located within the area defined by the parameters P-AXIS-00085 and P-AXIS-00093. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ReducedFeed-Zone
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 1 active, FALSE = Reduced velocity in zone 1 not active]
Redirection	
ST element	.X_Enable

Reduced velocity in zone 2, channel	
Description	When this signal is set, the path velocity is limited to the velocity defined in the axis parameter P-AXIS-00030 if the axis is located within the area defined by the parameters P-AXIS-00097 and P-AXIS-00105. If necessary the axis is decelerated after entering the zone.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ReducedFeed-Zone2
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Reduced velocity in zone 2 active, FALSE = Reduced velocity in zone 2 not active]
Redirection	
ST element	.X_Enable

Time override valid	
Description	Time override is activated
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_Override-TimeValid
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Time override is activated, time override is not activated]
Redirection	
ST element	.X_Enable

Time override	
Description	Time override can influence the internal CNC time base for motions. Its effect is similar to slow motion. Time override affects path velocity and acceleration differently. <u>Example:</u> If time override is 50% (command_w = 500) the velocity is reduced by a factor of 2 and acceleration by a factor of 4.
Data type	MCCControlUNS16Unit, see description of control unit
Special features	The channel parameter P-CHAN-00111 can be used to affect the effect of time override on the dwell time function. See safety not below.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlUNS16Unit_OverrideTime
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Unit	0,1 %
Data type	UINT
Value range	[100, 1000]
Redirection	
ST element	.X_Enable



Attention

Time override affects real-time and safety functions.

Delays in the use of these functions may lead to longer reaction times for safety functions under certain circumstances.

Time override affects path feed and acceleration independent of real-time functions such as feedhold or safety functions such as reduced speed. The user must consider this when using this function.

Disabling the detection of tool life data	
Description	Detecting tool life data is suppressed when this control unit is activated.
Data type	MCControlDistCtrlUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_ToolLifeSup-pressCapture
Commanded, requested and return value	
ST element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE = Tool life data is not detected, FALSE = Tool life data is detected]
Redirection	
ST element	.X_Enable

Delete distance to go, command	
Description	The rising edge of the commanded value has the effect that the CNC channel is decelerated to feed velocity 0. Then a linear motion is executed to the target position of the next motion block (short cut). The command only affects motion blocks. The functional description [FCT-C28] deals with the topic of "Delete distance to go" in detail.
Data type	MCControlBoolUnit, see description of Control Unit
Peculiarities	See state data X_DeleteDistanceToGoActive [► 88]
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlBoolUnit_DeleteDistanceToGo
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State (TRUE indicates that the command was detected by the CNC)
Data type	BOOL
Value range	[TRUE = distance to go to be deleted, FALSE = no impacts on motion blocks]
Redirection	
ST Element	.X_Enable

Delete distance to go, end marker	
Description	This control unit sets the end marker defined in the NC program online to valid. A bit mask is defined for this in the NC program. If at least one single bit in the end marker is set on the NC interface, this marker is valid as a jump target. Example: <pre>N10 #DEL DIST2GO [END = '16#0014']</pre> The end marker can be set to valid by bit3 (hexadecimal 4) or by bit5 (hexadecimal 0x10).
Data type	MC_CONTROL_UN32_UNIT, see description of control unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCControlBahn_Data.MCControlUNS32Unit_DeleteDistanceToGoActivation
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	UDINT
Value range	32-bit
Redirection	
ST Element	.X_Enable

Online tool compensation (OTC)	
Description	The tool radius can be adapted by setting the wear offset depending on wear. [0,1µm] See also Wear compensation of tool radius. It is only possible to use the control unit in the RADIUS or TOOL_DIR modes and in the DISC or AUTO mode. See NC command #OTC. See also functional description [FCT-C20].
Data type	MCCControlSGN32Unit, see description of Control Unit
Peculiarities	The wear offset is not output in a cycle in the CNC. Instead it is output over several cycles.
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlSGN32Unit_OTCRadiusOff-set
Commanded, requested and return values	
ST Element	.D_Command .D_Request .D_State
Data type	DINT
Value range	[-P-TOOL-00031, P-TOOL-00031]
Redirection	
ST Element	.X_Enable

Backward motion	
Description	Select/deselect backward motion on the path
Data type	MCCControlBoolUnit, see description of Control Unit
Access	SPS reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^^.addr^.MCCControlBahn_Data.MCCControlBoolUnit_BackwardMotion
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	.X_Enable

Simulated forward motion	
Description	Select/deselect simulated forward motion on the path For example, M function synchronisations are treated differently.
Data type	MCCControlBoolUnit, see description of Control Unit
Access	PLC reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCCControlBahn_Data.MCCControlBoolUnit_SimulateMotion
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	.X_Enable

Reset backward motion memory	
Description	Deselects backward motion memory No further NC block is saved in the memory. The memory is deleted.
Data type	MCCControlBoolUnit, see description of Control Unit
Access	SPS reads Request + State and writes Command + Enable
ST Path	pMC[channel_idx]^addr^.MCCControlBahn_Data.MCCControlBoolUnit_ResetBackward-Storage
Commanded, requested and return values	
ST Element	.X_Command .X_Request .X_State
Data type	BOOL
Value range	[TRUE, FALSE]
Redirection	
ST Element	.X_Enable

5 PLC

5.1 Control commands to PLC

5.1.1 Reset

PLC reset, axis	
Description	The PLC can be requested to perform a reset via this axis-specific interface. Here, the PLC must indicate that it want to be informed of reset requests by the CNC kernel by setting the X_Enable element.
Data type	MC_CONTROL_BOOL_UNIT, see description of control unit
Special features	Consumption data item
Access	CNC sets X_Command to TRUE to command a reset for the PLC. The CNC sets X_Command to FALSE after the PLC acknowledges execution of the rest by the X_State element.
ST path	pAC[axis_idx]^^.addr^.LcControlIpo_Data.LCControlBoolUnit_PLCReset
Commanded value	
ST element	.X_Command
Signal flow	CNC → PLC
Data type	BOOL
Value range	[TRUE = Reset request from CNC to PLC, FALSE]
Return value	
ST element	.X_State
Signal flow	PLC → CNC
Data type	BOOL
Value range	[TRUE = PLC executed reset, FALSE]
Requirement	
ST element	.X_Enable
Signal flow	PLC → CNC
Data type	BOOL
Value range	[TRUE = PLC wants to be notified about requests by CNC, FALSE]

PLC reset, channel	
Description	The PLC can be requested to execute a reset via this channel-specific interface. Here, the PLC must indicate that it wants to be informed of reset requests by the CNC kernel by setting the X_Enable element.
Data type	MC_CONTROL_BOOL_UNIT, see description of control unit
Peculiarities	Consumption data item
Access	CNC sets X_Command to TRUE to command a reset for the PLC. The CNC sets X_Command to FALSE after the PLC acknowledges the execution of the reset via the X_State element.
ST Path	pMC[channel_idx]^addr^.LCControlBahn_Data.LCControlBoolUnit_PLCReset
Commanded value	
ST Element	.X_Command
Signal flow	NCK → PLC
Data type	BOOL
Value range	[TRUE = Reset request from NCK to PLC, FALSE]
Return value	
ST Element	.X_State
Signal flow	PLC → NCK
Data type	BOOL
Value range	[TRUE = PLC executed reset, FALSE]
Request	
ST element	.X_Enable
Signal flow	PLC → NCK
Data type	BOOL
Value range	[TRUE = PLC wants to be notified about requests by CNC, FALSE]

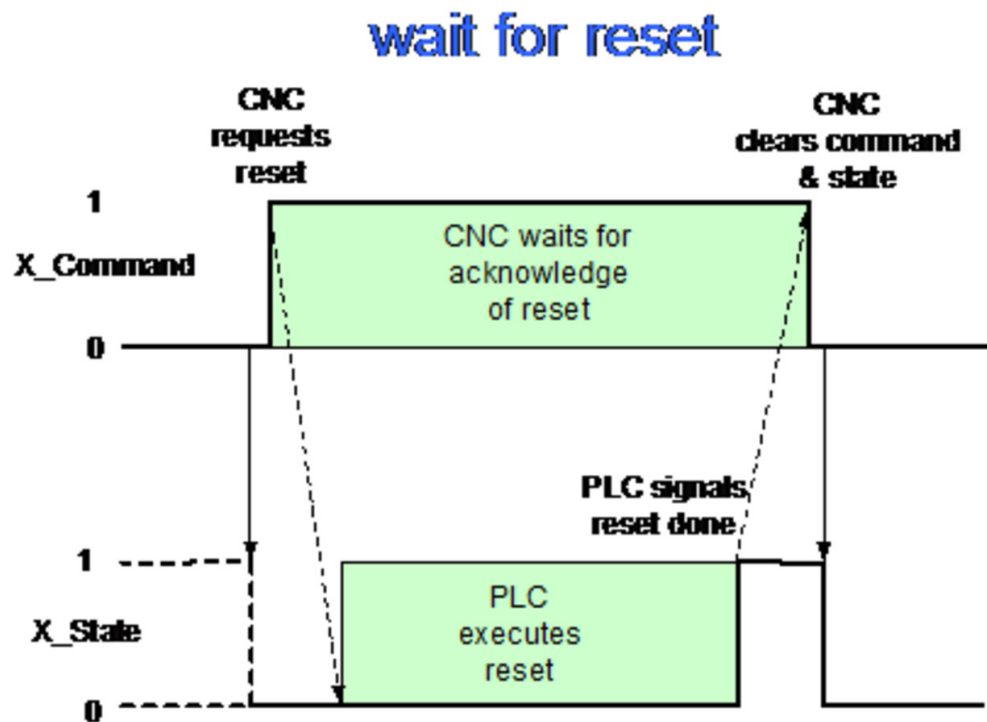


Fig. 17: Interaction between BOOLEAN-MC control unit and PLC



Notice

Before a new request and after detecting the reset acknowledgement, the CNC clears the X_State signal.

5.1.2

Block search

Block search on/off to PLC	
Description	
Data type	MC_CONTROL_BOOL_UNIT, see description of control unit
Special features	Consumption data item
Access	The NC kernel sets X_Command to TRUE when block search is to be enabled and then waits for acknowledgement from the PLC. The PLC acknowledges block search enable by setting the element X_State. Consequently, the NC kernel sets the X_Command to FALSE when block search is disabled and then waits for acknowledgement from the PLC which indicates this by cancelling the element X_State.
ST path	pMC[channel_idx]^addr^.LCControlBahn_Data.LCControlBoolUnit_BlockSearch
Commanded value	
ST element	.X_Command
Signal flow	CNC → PLC
Data type	BOOL
Value range	[TRUE = Block search is to be enabled, FALSE]
Return value	
ST element	.X_State
Signal flow	PLC → NCK
Data type	BOOL
Value range	[TRUE = PLC enabled block search, FALSE]
Requirement	
ST element	.X_Enable

wait for block skip on / off

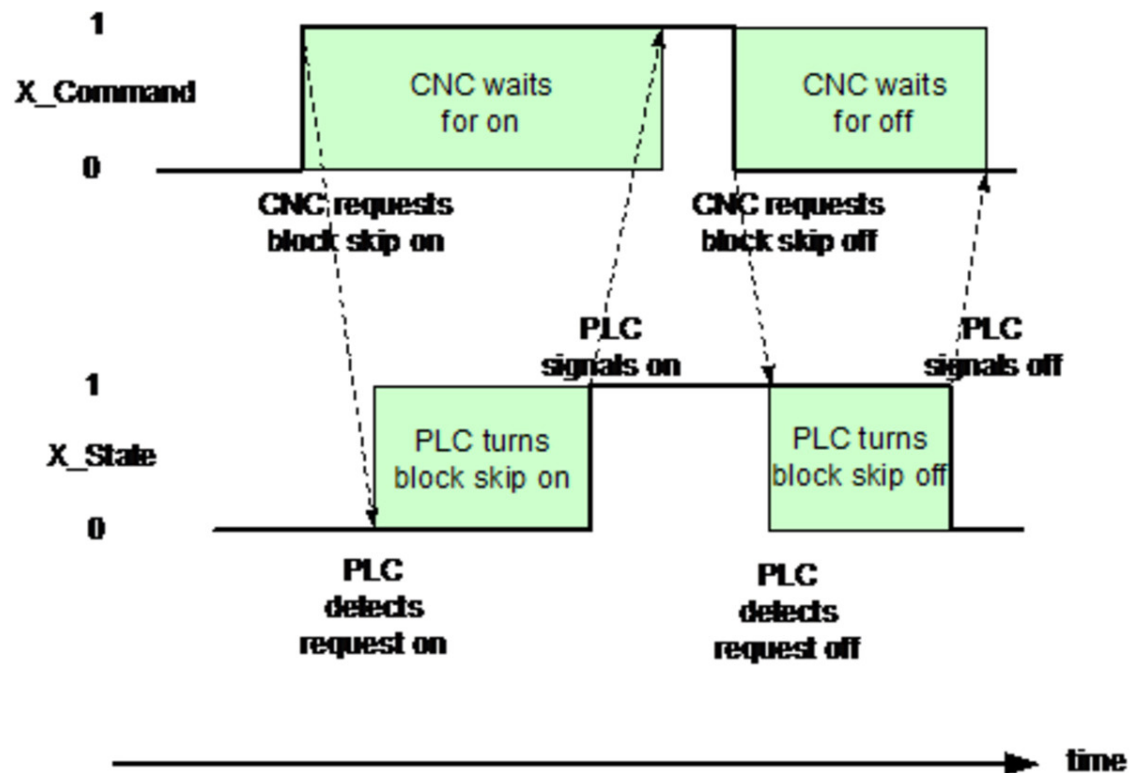


Fig. 18: Interaction between BOOLEAN-MC control unit and PLC



Notice

When the CNC is reset, the CNC resets **X_Command** and **X_State**.

6 Technology processes

6.1 Introduction

Technology functions can be defined for each channel and each axis.

Channel-specific technology functions are defined in the channel parameters; axis-specific technology functions are defined in the axis parameters of each axis.

This definition also includes the synchronisation mechanism of the technology function. A distinction is made between two basic types of synchronisation:

- Block-by-block synchronisation (default synchronisation),
- Cross-block synchronisation.

M functions not defined are indicated as unknown M functions by an error message after NC program start and decoding is aborted.

6.2 Management of technology functions

A basic distinction is made between two types of technology functions:

technology functions synchronised block-by-block and cross-block synchronised technology functions. This subdivision is also reflected on the High-Level Interface.

All technology functions to be synchronised block-by-block must be acknowledged at the latest at the end of the NC block in which they were programmed. They are therefore saved in consecutive sequence in the corresponding management field.

This is not the case with technology functions with cross-block synchronisation. In this case, the non-acknowledged technology functions yet to be executed may be distributed over the entire field (with gaps). In addition, it may occur that several identical technology functions are present in this field since, owing to cross-block synchronisation, the individual technology commands do not need to be acknowledged at the end of the NC block in which they were programmed. This must be allowed for at the PLC end to generate the acknowledgement of technology functions.

The number of technology functions programmed in an NC block is made available on the HLI for the array of the technology functions synchronised block-by-block. The number of non-acknowledged technology functions is specified for the array of cross-block synchronised technology functions.

6.3 Elements to manage axis-specific technology control units

6.3.1 Block-by-block synchronisation (default synchronisation)

Array of technology functions with block-by-block synchronisation	
Description	Array of M/H/S/T technology functions with block-by-block synchronisation. Technology functions are saved contiguously in this array.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.ATechnoUnitAxe_Std[tech_unit_idx]
Data type	ARRAY [1..HLI_MaxTechnoUnitsAxeStdSync] OF TechnoUnitAxe

Number of technology functions with block-by-block synchronisation	
Description	Number of valid entries in the array ATechnoUnitAxe_Std(= number of technology functions to be acknowledged in this block)
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.W_UsedUnitsStdSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsAxeStdSync]
Access	PLC reads

6.3.2 Cross-block synchronisation

Technology functions with cross-block synchronisation	
Description	Array of M/H/S/T technology functions with cross-block synchronisation. There may be entries of already acknowledged technology functions among the entries for non-acknowledged M functions.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.ATechnoUnitAxe_Late[tech_unit_idx]
Data type	ARRAY [1..HLI_MaxTechnoUnitsAxeLateSync] OF TechnoUnitAxe

Number of technology functions with cross-block synchronisation	
Description	Number of non-acknowledged technology functions in the array AtechnoUnitAxe_Late
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.W_UsedUnitsLateSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsAxeLateSync]
Access	PLC reads

6.4 Elements for managing channel-specific technology control units

6.4.1 Block-by-block synchronisation (default synchronisation)

Array of technology functions with block-by-block synchronisation	
Description	Array of M/H/S/T technology functions with block-by-block synchronisation. Technology functions are saved contiguously in this array.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx]
Data type	ARRAY [1..HLI_MaxTechnoUnitsChStdSync] OF TechnoUnitChannel

Number of technology functions with block-by-block synchronisation	
Description	Number of entries in the array ATechnoUnitChannel_Std (= number of technology functions to be acknowledged in this block)
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.W_UsedUnitsStdSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsChStdSync]
Access	PLC reads

6.4.2 Cross-block synchronisation

Technology functions with cross-block synchronisation	
Description	Array of M/H/S/T technology functions with cross-block synchronisation. There may be entries of already acknowledged technology functions among the entries for non-acknowledged M functions.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx]
Data type	ARRAY [1.. HLI_MaxTechnoUnitsChLateSync] OF TechnoUnitChannel

Number of technology functions with cross-block synchronisation	
Description	Number of entries in the array ATechnoUnitChannel_Late.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.W_UsedUnitsLateSync
Data type	UINT
Value range	[0, HLI_MaxTechnoUnitsChLateSync]
Access	PLC reads

6.5 Data of a technology control unit

6.5.1 Data of an axis-specific technology control unit

Data of a technology function, axis			
Description	A technology control unit contains elements for commanding, acknowledging and transferring any required parameters.		
Data type	TechnoUnitAxe		
ST path	Default synchronisation: pAC[axis_idx]^addr^.ATechnoUnitAxe_Std[tech_unit_idx] Cross-block synchronisation: pAC[axis_idx]^addr^.ATechnoUnitAxe_Late[tech_unit_idx]		
Job			
ST element	.X_Please		
Description	By setting X_Please, the CNC signals to the PLC that the technology control unit is to be executed.		
Data type	BOOL		
Value range	[TRUE, FALSE]		
Special features	Consumption data item		
Access	The CNC refreshes the data of the technology function only if this element is FALSE. After refreshment, the CNC sets this element to TRUE after the element X_Done was set to FALSE. The PLC reads the data of the technology function if this element has the value TRUE. After the data is transferred, the PLC sets the value to FALSE.		
Function type			
ST element	.W_FktCtrl		
Description	The technology function type is transferred in W_FktCtrl.		
Data type	UINT		
Value range	Value	Constant	Meaning
	1	GCW_250_HLIIntfMFkt	M function
	2	GCW_250_HLIIntfHFkt	H function
	3	GCW_250_HLIIntfSpindel	S function
Access	PLC reads		
Parameter			
ST element	.MSTHProsessAxe_Attribut		
Description	Depending on the contents of the element W_FktCtrl, this element contains the parameters of an M function/H function if the technology function type is GCW_250_HLIIntfMFkt or GCW_250_HLIIntfHFkt S function (spindle) if the technology function type is GCW_250_HLIIntfSpindel		
Data type	ARRAY [1.. HLI_TechnoUnionByteCountAxe] OF BYTE		
Access	PLC reads		
Acknowledgement			

ST element	.X_Done
Description	By setting the element X_Done to TRUE, the PLC signals to the CNC that the technology information was executed.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	PLC sets the value to TRUE when the technology function was executed. The CNC sets the value to a FALSE before a new command.

6.5.2 Data of a channel-specific technology control unit

Data of a technology function, channel			
Description	A technology control unit contains elements for commanding, acknowledging and transferring any required parameters.		
Data type	TechnoUnitChannel		
ST path	Default synchronisation: pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx] Cross-block synchronisation: pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx]		
Job			
ST element	.X_Please		
Description	By setting X_Please, the CNC signals to the PLC that the technology control unit is to be executed.		
Data type	BOOL		
Value range	[TRUE, FALSE]		
Special features	Consumption data item		
Access	The CNC refreshes the data of the technology function only if this element is FALSE. After refreshment, the CNC sets this element to TRUE after the element X_Done was set to FALSE. The PLC reads the data of the technology function if this element has the value TRUE. After the data is transferred, the PLC sets the value to FALSE.		
Function type			
ST element	.W_FktCtrl		
Description	The technology function type is transferred in W_FktCtrl.		
Data type	UINT		
Value range	Value	Constant	Function
	1	HLI_INTF_M_FKT	M function
	2	HLI_INTF_H_FKT	H function
	3	HLI_INTF_SPINDEL	S function
	4	HLI_INTF_TOOL	T function
Access	PLC reads		
Parameter			
ST element	.MSTHProsessChannel_Attribut		
Description	Depending on the contents of the element W_FktCtrl, this element contains the parameters of an M function/H function if the technology function type is GCW_250_HLIIntfMFkt or GCW_250_HLIIntfHFkt S function (spindle) if the technology function type is GCW_250_HLIIntfSpindel T function if the technology function type is GCW_250_HLIIntfTool		

Data type	ARRAY [1.. HLI_TechnoUnionByteCountChannel] OF BYTE
Access	PLC reads
Acknowledgement	
ST element	.X_Done
Description	By setting the element X_Done to TRUE, the PLC signals to the CNC that the technology information was executed.
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	PLC sets the value to TRUE when the technology function was executed. The CNC sets the value to a FALSE before a new command.

6.6 Data of the technology functions

6.6.1 Data of the M/H function

Data of the M function/H function	
Description	The function number and the execution time are transferred as additional parameters of an M or H function.
Data type	MHProzess
ST path	pMHProcess : POINTER TO MHProzess; Channel-specific, default synchronisation: pMHProcess := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx].MSTHProzessChannel_Attribut.AB_Data[1]); Channel-specific, cross-block synchronisation: pMHProcess := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx].MSTHProzessChannel_Attribut.AB_Data[1]);
Access	PLC reads
Number of the M function/H function	
Description	Number of the M or H function. This corresponds to the number for an M or H function programmed in the NC program. Example: For instance, if M4711 was programmed.
ST element	.D_Number
Data type	UDINT
Access	FktNr : UDINT; FktNr := pMHProcess^.D_Number
Probable execution time M/H function	
Description	Contains values configured for the channel parameters in the entry P-CHAN-00040 or P-CHAN-00026 (m_prozess_zeit[]). These values can be used to implement timeout monitoring for technology functions in the PLC application. or If synchronisation type MOS_TS is configured, the sample time offset of the M/H technology function is indicated in this element. It is calculated and output by the NC kernel.
ST element	.D_Time
Data type	UDINT
Access	Time : UDINT; Time := pMHProcess^.D_Time
Special features	Re. 1.): For spindle-specific M functions the corresponding channel parameter is called mX_prozess_zeit where [X = 3, 4, 5, 19]. Example: The parameter for the M3 function of a spindle is: spindel[index].m3_prozess_zeit. Bsp: The characteristic parameter for an M function is: spindel[index].mX_prozess_zeit.
Block number of the M/H function	
Description	Block number from the NC program to specify the program line in the NC program in which the M or H function was programmed.
ST element	.D_BlockNumber

Data type	UDINT
Access	BlockNr : UDINT; BlockNr := pMHProcess^.D_BlockNumber
Program line number of the M/H function	
Description	Number of the NC program line in which the M/H function was programmed.
ST element	.D_PrgLineNumber
Data type	UDINT
Access	PrgLineNr : UDINT; PrgLineNr := pMHProcess^.D_PrgLineNumber
Number as additional information	
Description	A number assigned to an M or H function in the NC program and transferred when the M/H function is output to the interface. See [PROG//M/H functions with additional information].
ST element	.D_AddNumber
Data type	DINT
Access	AddNr : DINT; AddNr := pMHProcess^.D_AddNumber
Number of cross-block synchronised technology functions not acknowledged by the PLC	
Description	Number of technology functions not yet acknowledged by the PLC but with cross-block synchronisation and waiting at the interface. The number contains all types of technology functions.
ST element	.W_NrLateSync
Data type	UINT
Access	NrLateSync : UINT; NrLateSync := pMHProcess^.W_NrLateSync

6.6.2 Data of the S function

When M functions (M03, M04, M05, M19) associated with a spindle are programmed, the technology function data is saved as an S function in the associated **axis-specific** HLI area.

Data of the M function/H function	
Description	The parameters of an S function are contained in the SProzess structure.
Data type	Sprozess
ST path	pSProcess : POINTER TO SProzess; Axis-specific, default synchronisation: pSProcess := ADR(pAC[axis_idx]^addr^.ATechnoUnitAxe_Std[tech_unit_idx].MSTH-ProzessAxe_Attribut.AB_Data[1]); Axis-specific, cross-block synchronisation: pSProcess := ADR(pAC[axis_idx]^addr^.ATechnoUnitAxe_Late[tech_unit_idx].MSTH-ProzessAxe_Attribut.AB_Data[1]);
Access	PLC reads
Command position of spindle with M19	
Description	Command position for spindle position with M19
ST element	.D_Pos
Data type	DINT
Unit	10 ⁻⁴ °
Access	ActivePosition : DINT; ActivePosition := pSProcess^.D_Pos;pMHProcessData^.D_Pos
Programmed spindle speed	
Description	Programmed spindle speed
ST element	.D_Rev
Data type	DINT
Unit	10 ⁻³ °/s
Access	PrgRevolution: DINT; PrgRevolution: = pSProcess^.D_Rev;
Probable execution time	
Description	Time probably required to process an S function.
ST element	.D_Zeit
Data type	UDINT
Unit	1 µs
Access	ExpectedTime : UDINT; ExpectedTime:= pSProcess^.D_Zeit;
Number of the M function of the spindle indexing function	
Description	Number of the spindle indexing function (M03, M04, M05)
ST element	.W_MoveCmd
Data type	UINT

Value range	Value	M function
	3	M03
	4	M04
	5	M05
Access	MoveCmdNum : UINT; MoveCmdNum := pSProcess^. W_MoveCmd ;	
Number of the M function of the spindle positioning function		
Description	Number of the spindle positioning function (M19)	
ST element	.W_PosCmd	
Data type	UINT	
Value range	19 stands for M19	
Access	PosCmdNum : UINT; PosCmdNum := pSProcess^. W_PosCmd ;	
Axis number		
Description	Unique system-wide number of a logical axis/spindle	
ST element	.W_log_AxeNr	
Data type	UINT	
Value range	TwinCAT PLC typically [1, gNrAx]	
Access	LogAxisNum : UINT; LogAxisNum := pSProcess^. W_Log_AxeNr ;	
Number of cross-block synchronised technology functions not acknowledged by the PLC		
Description	Number of technology functions not yet acknowledged by the PLC but with cross-block synchronisation and waiting at the interface. The number contains technology functions of all type.	
ST element	.W_NrLateSync	
Data type	UINT	
Access	NrLateSync : UINT; NrLateSync := pSProcess^. W_NrLateSync	

6.6.3 Data of the T function

Data of the T function	
Description	All data for a tool change is combined in a T function.
Data type	TProzess
ST path	pTProzess : POINTER TO TProzess; Channel-specific, default synchronisation: pTProzess := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Std[tech_unit_idx].MSTHProcessChannel_Attribut.AB_Data[1]); Channel-specific, cross-block synchronisation: pTProzess := ADR(pMC[channel_idx]^addr^.ATechnoUnitChannel_Late[tech_unit_idx].MSTHProsessChannel_Attribut.AB_Data[1]);
Access	PLC reads
Tool identification	
Description	The structure contains the identification number of the tool. There also may be identification numbers for identical or similar tools. A description of the structure format is contained in User data of tool identification [► 147]
ST element	.HLIToolID_Data
Data type	HLIToolId [► 146]
Access	HLIToolId : HLIToolId HLIToolId := pTProzess^.HLIToolID_Data;
Information on the tool unit	
Description	Information supplied to the PLC referring to a tool unit.
ST element	.D_AddInfo[]
Data type	ARRAY [1..HLI_MAX_NBR_ADD_INFO] OF HLI_UN32
Access	AddInfo: UDINT; AddInfo := pTProzess^.D_AddInfo[X] where X = [1, HLI_MAX_NBR_ADD_INFO]
Number of cross-block synchronised technology functions not acknowledged by the PLC	
Description	Number of technology functions not yet acknowledged by the PLC but with cross-block synchronisation and waiting at the interface. The number contains technology functions of all type.
ST element	.W_NrLateSync
Data type	UINT
Access	NrLateSync : UINT; NrLateSync := pTProzess^.W_NrLateSync

6.6.3.1 User data of tool identification

With the definition of pTProzess [► 146] from the previous section and the definition

pHLIToolId: POINTER TO HLI_TOOL_ID applies:

pHLIToolId := ADR(pTProzess^.id);

and is used as shown in the table below.

Number of the tool to be replaced	
Description	Number of the tool to be replaced
ST element	.D_Basic
Data type	DINT
Access	ToChangeToolNum : DINT; ToChangeToolNum := pHLIToolId^.D_Basic;
Number of a sister tool	
Description	Number of an identical sister tool
ST element	.D_Sister
Data type	DINT
Access	SisterToolNum : DINT; SisterToolNum := pHLIToolId^.D_Sister;
Number of a variant tool	
Description	Number of a similar variant tool
ST element	.D_Variant
Data type	DINT
Access	VariantToolNum : DINT; VariantToolNum := pHLIToolId^.D_Variant;
Sister tool valid	
Description	Validity identifier for the sister tool.
ST element	.X_SisterValid
Data type	BOOL
Access	SisterToolValid : BOOL; SisterToolValid := pHLIToolId^.X_SisterValid;
Variant tool valid	
Description	Validity identifier for the variant tool.
ST element	.X_VariantValid
Data type	BOOL
Access	VariantToolValid : BOOL; SisterToolValid := pHLIToolId^.X_VariantValid;

7 External variables / V.E variables

Data can be exchanged between the NC program and the PLC via the HLI using external variables. Each channel has a separate data area for external variables known only in the channel. In addition, there is a cross-channel global data area which all channels can access.

At the PLC end, the data areas for the external variables are represented as an ARRAY OF UDINT. The index of the individual array elements starts with the value 1.

Each external variable always occupies one memory block of HLI_VEByteCount (24) bytes, irrespective of its data type. If an array of external variables was defined, the individual variables are saved packed in the memory area (several variables per memory block). Here, several consecutive memory blocks may be used depending on the array size.

When the PLC accesses external variables, the index of the variables in the memory area of the external variables must first be determined:

Example for calculating an index:

4th variable (VarNr = 4):

Offset = (VarNr - 1) * HLI_VEByteCount / 4 + 1

A variable with the index = 3 results in an offset in the memory of 13.

The memory area must then be accessed depending on the actual data type of the external variables. All information required for this is contained in the configuration list of the external variables.

Please refer to the documentation [EXTV] for further details on external variables.

If the NC kernel runs in the TwinCAT runtime environment, the number of external variables can be configured in the System Manager.

An example program to access external variables is available under the name of HLI-Ve1.pro.

External variable	
Description	Memory area for data exchange between NC program and PLC
Data type	VeData
ST Path	PVeData : POINTER TO VEData; CNC global: pVeData := ADR(pVeGlob^.addr); channel-specific: pVeData := ADR(pVe[channelIdx]^ .addr);
Access	PLRealVal : POINTER TO LREAL; LRealVal : LREAL; VeOffset : DINT; VeOffset := (VarNr - 1) * HLI_VEByteCount / 4 + 1; pLRealVal := ADR(pVeData^.AHLI_UN32_Data[VeOffset]); LRealVal := pLRealVal^;
Peculiarities	Access depends on the data type of the external variables.

8 Processing types

The CNC distinguishes between 5 operation modes. Switchover between these operation modes is performed using the operator interface and/or PLC interface. However, only **one operation mode may be active at any one time**.

The following operation modes are defined:

Operation mode	ST constant	Value	Explanation
Standby	HLI_IMCM_STANDBY_MODE	1	No operation mode is selected. Default after controller start-up.
Automatic mode	HLI_IMCM_AUTOMATIC_MODE	2	The controller can process a complete NC program automatically. Program execution can be interrupted and resumed.
Manual block	HLI_IMCM_MDI_MODE	3	Motions are commanded by the operator computer via a single NC block. The NC block is transferred to the controller in a string and executed via a START command. A motion can be interrupted and resumed.
Manual operation mode	HLI_IMCM_MANUAL_MODE	4	Motions are commanded by peripherals connected directly to the controller (buttons, handwheels).
Homing	HLI_IMCM_REFERENCE_MODE	5	Axes can be referenced. An NC program with the name rpf.nc then starts.

An operation mode may have different states. The states of individual operation modes and their significance for the operation mode are discussed in the sections below.

8.1 State graph of operation modes

Note: There is no state graph for the “Standby” operation mode.

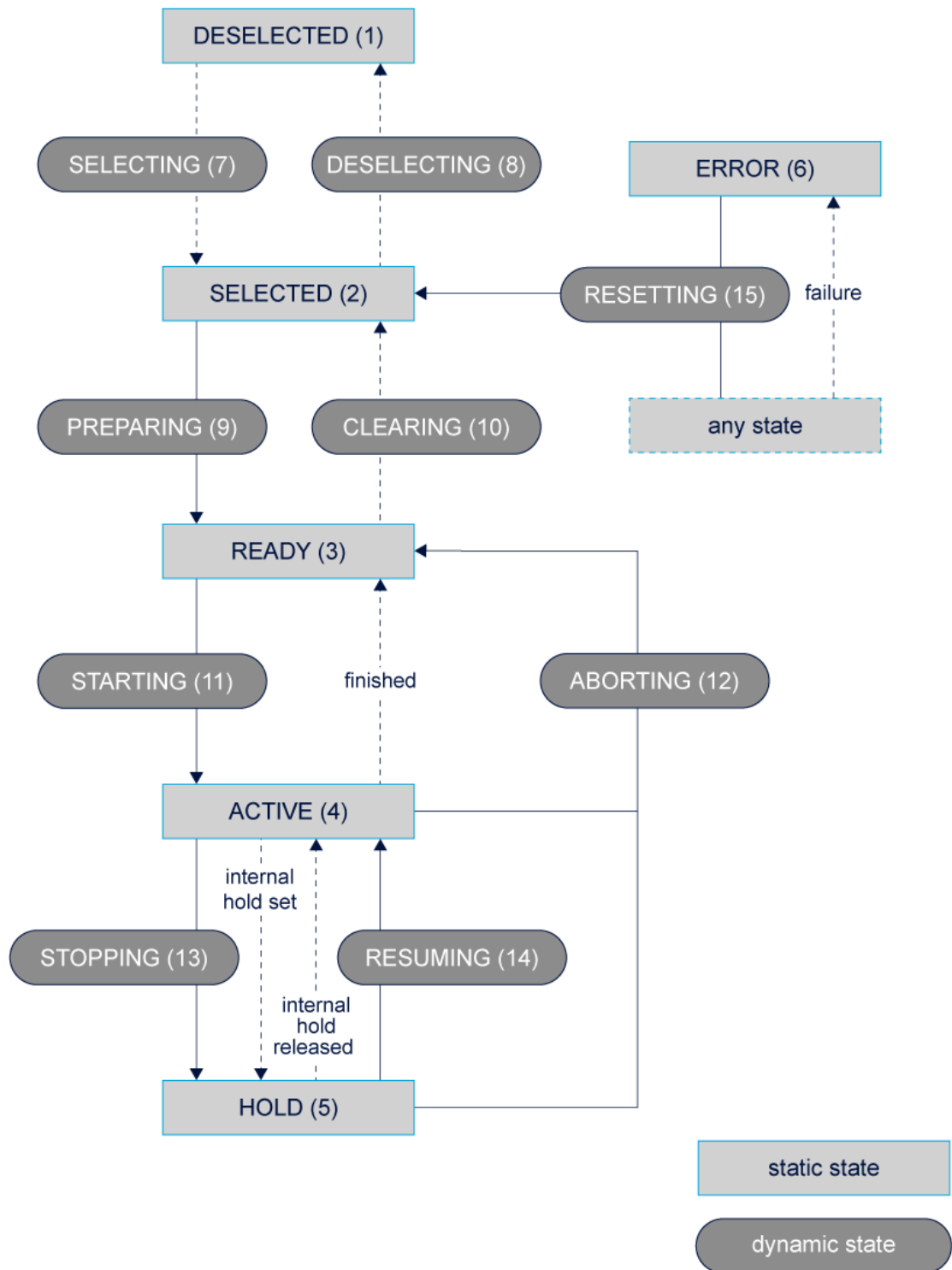


Fig. 19: State graph of an operation mode



Notice

A CNC reset is automatically executed when the target operation mode and target state are commanded if the state change specifies an abort (see “abort”).

The error state is exited by an automatically executed reset if an operation mode was commanded.

In addition, a CNC reset can be forced by the following commands:

- a) Select operation mode = STANDBY and state = SELECTED
- b) Explicit specification of transition = RESET in the currently active operation mode
- c) Specify the target state = RESETTNG (see below)

8.1.1 States of operation mode: Automatic mode

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Operation mode is deselected
SELECTED	GCW_250_HLIImcmProcessSelected	2	Automatic mode is selected
READY	GCW_250_HLIImcmProcessReady	3	NC program is selected
ACTIVE	GCW_250_HLIImcmProcessActive	4	NC program is running
HOLD	GCW_250_HLIImcmProcessHold	5	NC program is interrupted (see also Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	An error occurred while the NC program is executed.

States of Automatic mode

The NC program name must be transferred where there is a transition from selected to ready.

8.1.2 States of operation mode: Manual block

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Manual block mode is deselected.
SELECTED	GCW_250_HLIImcmProcessSelected	2	Manual block mode is selected. An NC block (NC blocks) can be programmed.
READY	GCW_250_HLIImcmProcessReady	3	??? MDI block(s) is/are selected. ??? CNC accepted the programmed NC block(s).
ACTIVE	GCW_250_HLIImcmProcessActive	4	The NC block(s) is/are running.
HOLD	GCW_250_HLIImcmProcessHold	5	NC block(s) is/are stopped (see also feedhold). Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	Error state

States of Manual block mode

The manual block (string) must be transferred when there is a transition from SELECTED to READY.

8.1.3 States of operation mode: Manual operation mode

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Operation mode is deselected.
SELECTED	GCW_250_HLIImcmProcessSelected	2	Operation mode is selected. (Basic state).
READY	GCW_250_HLIImcmProcessReady	3	Manual mode is programmed.
ACTIVE	GCW_250_HLIImcmProcessActive	4	Manual mode is running.
HOLD	GCW_250_HLIImcmProcessHold	5	Manual mode is stopped (see also feedhold). Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	Error state

States of Manual mode

8.1.4 States of operation mode: Homing

State	ST constant	Value	Description
DESELECTED	GCW_250_HLIImcmProcessDeselected	1	Operation mode is deselected.
SELECTED	GCW_250_HLIImcmProcessSelected	2	Operation mode is selected. (Basic state).
READY	GCW_250_HLIImcmProcessReady	3	Homing is programmed (e.g. with respect to sequence).
ACTIVE	GCW_250_HLIImcmProcessActive	4	Homing is running.
HOLD	GCW_250_HLIImcmProcessHold	5	Homing is stopped (see also feedhold). Feedhold).
ERROR	GCW_250_HLIImcmProcessError	6	Error state

States of Homing mode

Axes to be referenced must be transferred during the transition from SELECTED to READY. If no string is transferred, the default homing program rpf.nc is started.

8.2 Control commands/status information for operation modes

The control unit described below contains data to command a change of operation mode and to poll the current state of operation mode management.

8.2.1 Control Unit

Operation mode	
Description	Control unit to switch over the operation mode and poll the current state of operation mode management, including flow control of user data.
Data type	MCCControlMCMMode_State_Unit, see description of control unit with usage check
ST path	pMC[channel_idx]^addr^.MCCControlChannel_Data.MCCControlMCMMode_State_Unit
Commanded, requested data	
ST element	.HLIProcTransTo_Mode_StateCommand .HLIProcTransTo_Mode_StateRequest
Data type	HLIProcTransTo_Mode_State [► 155]
Access	PLC writes Command and reads Request
Return data	
ST element	.HLI_ElmCmProcState_ModeStateRequest
Data type	HLI_ElmCmProcState_ModeState [► 159]
Access	PLC reads
Flow control of commanded value	
ST element	.X_McmCommandSemaphor
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_McmRequestSemaphor
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST path	.X_McmEnable

8.2.2 User data

8.2.2.1 Requested and commanded user data

Initial operation mode		
Description	Operation mode from which the operation mode is to be changed.	
ST path	Commanded value pMC[channel_idx]^^.addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand Requested value pMC[channel_idx]^^.addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest	
Commanded, requested value		
ST element	.X_McmCommandFromMode	
Data type	UDINT	
Value range	Value	Constant
	1	GCW_250_HLIImcmStandbyMode
	2	GCW_250_HLIImcmAutomaticMode
	3	GCW_250_HLIImcmMDIMode
	4	GCW_250_HLIImcmManualMode
	5	GCW_250_HLIImcmReferenceMode
Special features	This element does not need to be programmed when the operation mode is changed. However, if a value is specified, a check is conducted when the operation mode is changed to establish whether the CNC is actually in the specified operation mode. A warning is issued if this is not the case.	

Initial state of operation mode when the operation mode is changed		
Description	State within the operation mode from which the state switchover is to occur.	
ST path	Commanded value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand Requested value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest. X_McmCommandFromState	
Commanded, requested value		
ST element	.X_McmCommandFromState	
Data type	UDINT	
Value range	Value	Significance / constant
	0	no significance
	1	GCW_250_HLIImcmProcessDeselected
	2	GCW_250_HLIImcmProcessSelected
	3	GCW_250_HLIImcmProcessReady
	4	GCW_250_HLIImcmProcessActive
	5	GCW_250_HLIImcmProcessHold
	6	GCW_250_HLIImcmProcessError
Special features	This element does not need to be programmed when the operation mode is changed. However, if a value is specified, a check is conducted when the operation mode is changed to establish whether the CNC is actually in the specified operation mode. A warning is issued if this is not the case.	

Target operation mode when the operation mode is switched over	
Description	Operation mode to which the system is to switch.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand. Requested value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest.
Commanded, requested value	
ST element	.X_McmCommandToMode
Data type	UDINT
Value range	See "Operation mode actual value on of operation mode change" → Value range

Target state when operation mode is changed	
Description	Target state within the target operation mode.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand Requested value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest
Commanded, requested value	
ST element	.X_McmCommandToState
Data type	UDINT
Value range	See "State actual value of operation mode at operation mode change" → value range



Release Note

If the target state = RESETTING = 15 is specified, a CNC reset is executed explicitly.

This function is available as of the following Build:

V2.10.1033.01 and higher

V2.10.1507.02 and higher

V2.10.1800.04 and higher

Parameters for operation mode change	
Description	Parameters for operation mode change. It may be necessary to specify parameters when commanding an operation mode change to ensure the successful change to a specific state of an operation mode. These parameters are saved in this element.
ST Path	pParameter: POINTER TO STRING; Parameter : STRING; Commanded value pParameter := ADR(pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand.AB_Data[1]); Requested value pParameter := pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateRequest.AB_Data[1] Parameter := pParameter^;
Data type	ARRAY[1..GCW_250_HLIImCmProcModeStateParaSize] OF BYTE
Peculiarities	The length of the character string is limited to maximum 83 characters for AUTOMATIC operation mode. If the character string is longer, it is recommended to use the parameterisation of file paths. See P-CHAN-00401to P-CHAN-00404.

If an operation mode change is commanded, it may be necessary to specify a parameter so that commanding can be executed successfully. The table below lists the cases where this is necessary and what type of parameter needs to be transferred. If one of the listed state transitions is involved in the operation mode and state change, the corresponding parameter must be specified. Consider the "State graph of other operation modes" to determine whether one of the state transitions listed below is executed in case of a commanded operation mode change.

Parameters for operation mode change

The content of the "parameter" structure element is only adopted by the NC kernel if a state transition is commanded from SELECTED to one of the states READY, ACTIVE or HOLD.

Operation mode setpoint	Parameter
Automatic mode	The NC program name as character string.
Manual block	NC block (blocks)
Manual mode	no parameter → all axes are activated (G200) explicit activation of specific axes with G200[Axis_1, ...]
Homing	no parameter → NC program rpf.nc is started explicit selection of axes by manual block (e.g.: G74 X1 Z2)

Channel number	
Description	Number of the channel whose operation mode is to be switched over.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLIProcTransTo_Mode_StateCommand.X_McmCommandChannelNo Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request.X_McmCommandChannelNo
Data type	UDINT
Special features	Not used (only for compatibility with the HÜMNOS standard).

8.2.2.2 Status information

Operation mode actual value	
Description	Actual value of the operation mode.
ST path	pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLI_ElmcProcState_ModeStateRequest.X_McmRequestReadMode
Data type	UDINT
Value range	See “Operation mode actual value on of operation mode change” → Value range

State actual value of the operation mode.	
Description	Actual value of the operation mode state.
ST path	pMC[channel_idx]^addr^.MCControlChannel_Data.MCControlMCM-Mode_State_Unit.HLI_ElmcProcState_ModeStateRequest.X_McmRequestReadState
Data type	UDINT
Value range	See “State actual value of operation mode at operation mode change” → value range

9 Manual operation mode

The manual mode function permits the external control of individual axes with physical elements of manual mode (handwheel, continuous jog buttons) via the HLI.

The following three options are available for axis motion:

- **Handwheel function:** : any path at any velocity by specifying handwheel increments.
- **Cont. jog mode:** any path at defined velocity when a button is operated.
- **Incr. jog mode:** defined path at defined velocity when a button is operated.

It is possible to assign the manual mode elements to one single or multiple axis and to change the parametrisation (e.g. the incremental jogging distance) during operation. The control element is assigned to the logical axes via the logical axis number. The diagram below shows as an example of an assignment of manual mode elements to CNC axes.

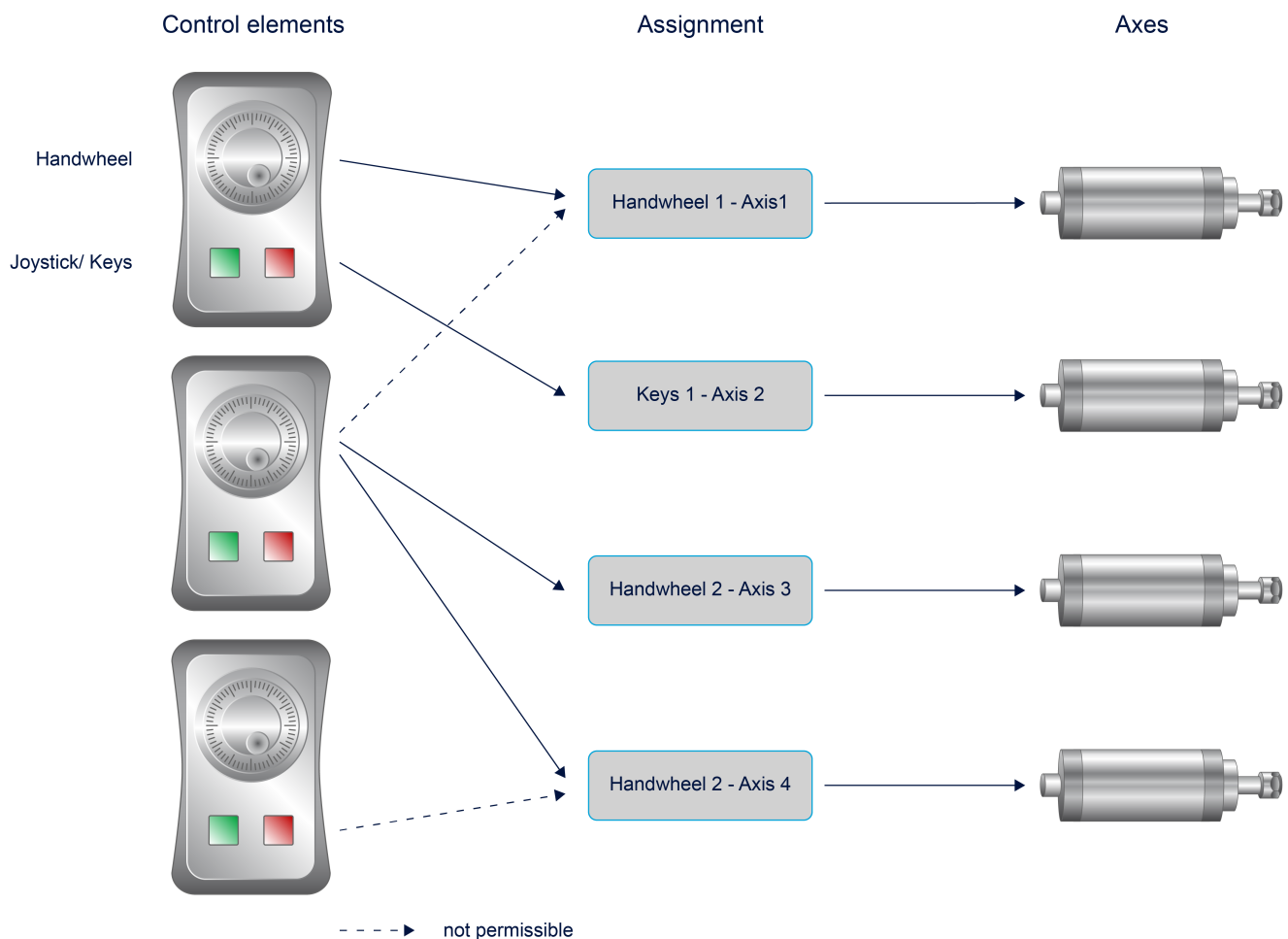


Fig. 20: Control elements and assignment

In manual mode, each axis can assume three different states:

- IDLE
- READY
- ENABLED

The following steps are required to use manual mode for an axis:

- Activating the READY state for axes

- Parametrising the manual mode type
- Activating manual mode (transition to ACTIVE state)
- Transferring control element actions (button presses, handwheel counter state) to the CNC.

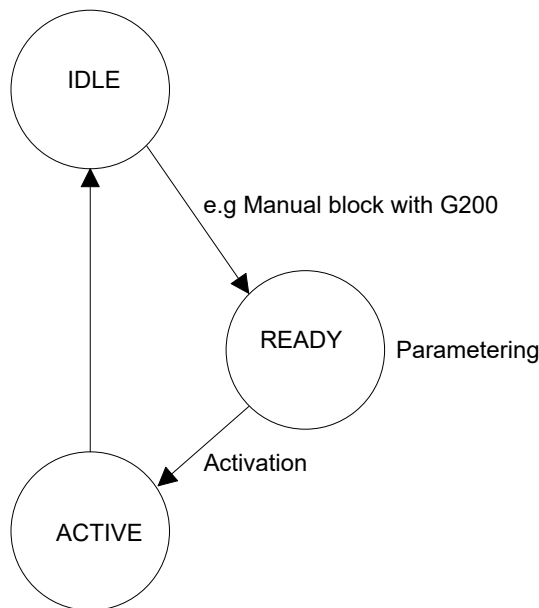


Fig. 21: Manual mode state transitions

Transition to READY state

The transition to READY state can be executed by the following actions:

- Explicit switchover of operation mode via HLI or GUI
- Programming of G200/G201 in the NC program or manual block

The desired manual mode type can then be parameterised in the READY state. See also the section Parameterising manual mode

Transition to READY state

The axis transits to the ACTIVE state as soon as a control element was assigned. Details are contained in the section Activating control elements.

Transferring control element actions

In the ACTIVE state, control element actions can now be transferred to the CNC to move the axis.

Deselecting manual mode

The ACTIVE state of an axis is deselected when the axis is assigned to the control element 0 or If a reset was executed.

Information on the axis state regarding manual mode is contained in the structure HLI_HB_AXIS_DISPLAY_DATA (see Section Status information of manual mode).

9.1 Status information of manual mode

State of manual mode		
Description	The manual operation mode is in one of the states described below.	
Signal flow	CNC → PLC	
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. W_State	
Data type	UINT	
Value range	Value	Meaning
	0	Operation mode deactivated
	1	Operation mode enabled in NC program via G200, G201 but not assigned to a control element.
	2	Operation mode enabled in NC program via G200, G201, assigned to control element.
Access	PLC reads	

Operation mode of manual mode		
Description	Look ahead waits to receive request from an axis.	
Signal flow	CNC → PLC	
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. W_OperationMode	
Data type	UINT	
Value range	Value	Operation mode
	0	no operation mode selected
	1	Handwheel mode
	2	Continuous jog mode
	3	jog mode
Access	PLC reads	

Control element number	
Description	Logical number of the control element currently linked to the axis in question.
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. W_ControlElement
Data type	UINT
Access	PLC reads

Handwheel resolution	
Description	Resolution of the handwheel which is linked as control element to the axis in question.
Signal flow	CNC → PLC
Unit	Increments per handwheel pulse
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. D_HRAufloesung
Data type	DINT
Access	PLC reads

Path velocity in continuous jog mode	
Description	Path velocity of the axis in question when moved in continuous jog mode.
Signal flow	CNC → PLC
Unit	mm/min
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. D_TippGeschw
Data type	DINT
Access	PLC reads

Path velocity in incremental jog mode	
Description	Path velocity of the axis in question when moved in continuous jog mode.
Signal flow	CNC → PLC
Unit	mm/min
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data. D_JogGeschw
Data type	DINT
Access	PLC reads

Motion path in incremental jog mode	
Description	Motion path of the axis in question for each button press when moved in incremental jog mode.
Signal flow	CNC → PLC
Unit	Mm
ST path	pMC[channel_idx]^addr^.StateBahn_Data.HLIBahnCoordDispData_Coord[axis_idx].AxeHBDisplayData_Data.D_JogWeg
Data type	DINT
Access	PLC reads

9.2 Control commands of manual mode

The individual manual operation modes are controlled by specific control units. These control units contain data on the flow control of user data and the user data itself. User data generally comprises elements of a structure.

9.2.1 Activating control elements for manual mode

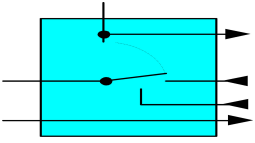
After an axis was prepared for manual mode by G200/G201, this axis can be assigned a logical control element (button/handwheel) by a parameter set. This takes place during activation of an axis.

The default value settings are defined in the axis-specific parameter lists:

```
# 1 handwheel, 2 continuous, 3 incremental
handbetrieb.default.operation_mode 2 # 2 continuous,
handbetrieb.default.control_element 1 # logical handwheel/key
```

The 0th parameter set (index = 0) of the manual mode parameters is used as default parameter.

Every time the manual mode (see Operation mode types) or G200/G201 is reselected, the last known setting of the axes (operation mode, link to control element and parameter set) is restored.

Activating control elements with manual mode	
Description	Control unit to manage data to activate a control element and assign it to an axis in manual mode, including flow control of user data.
 Data type	HBAutomationControlUnit, see description Control unit with usage check
Special features	This control unit can only be used if the CNC is in manual mode or if the axes are enabled for manual mode by an explicit G200/G201. Otherwise, activation is refused with an error message (e.g. 150048 -> "Impermissible selection of operation mode in continuous jog mode").
ST path	pMC[channel_idx]^addr^.MCControlHB_Data.HBAutomationControlUnit_Activation
Access	PLC reads HBAutomation_Request and writes HBAutomation_Command + X_Enable
Commanded, requested data	
ST element	.HBAutomation_Command .HBAutomation_Request
Data type	HBAutomation [► 167], for description see Section User data on activation [► 166]
Flow control commanded data	
ST element	.X_CommandSemaphor
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_RequestSemaphor
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST path	.X_Enable

9.2.1.1 User data on activation

Axis number	
Description	Unique system-wide number of a logical axis. An control element is assigned to the specified logical axis with which the axis is to be moved in manual mode.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command.W_LogAchsNr Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request.W_LogAchsNr
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Control element number	
Description	Number of the logical control element to be assigned to the logical axis.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command.W_ControlElement Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request.W_ControlElement
Data type	UINT
Value range	When continuous and incremental jog mode is activated: one of the values defined as logical button pair numbers in the configuration list hand_mds.lis for the characteristics tasten_data[X].log_tasten_nr. When handwheel mode is activated: one of the values defined as logical handwheel numbers in the configuration list hand_mds.lis for the characteristics hr_data[0].log_hr_nr.
Special features	If 0 is specified as the logical number, the current operation mode of an axis is deselected.

Manual operation mode		
Description	Manual operation mode to be assigned to the logical axis.	
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command. W_OperationMode Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request. W_OperationMode	
Data type	UINT	
Value range	Value	Operation mode
	0	no operation mode, current operation mode selected
	1	Handwheel mode
	2	Continuous jog mode
	3	jog mode

Manual operation mode parameter set		
Description	Specifies the index of the parameter set to be used for the manual mode.	
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Command. W_ParameterIndex Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBActivationControlUnit_Activation.HBActivation_Request. W_ParameterIndex	
Data type	UINT	
Value range	[0; 2]	
Special features	The first value set in the parameter table (index = 0) is overwritten by the PLC interface when individual parameters are specified. The remaining parameter sets are not changed. This means, they correspond to the values specified in the axis-specific parameter lists.	

Remark:

The currently active axis can be deactivated by deselecting the operation mode or by assigning the control element 0.

9.2.2 Parametrising of manual mode

Basically, every manual mode can be specifically parametrised. For example, the motion velocities or incremental step can be defined per axis. The default values here are pre-assigned in the axis-specific parameter lists.

handbetrieb.hr.aufl[1]	20
handbetrieb.hr.aufl[2]	30
#	
handbetrieb.tipp.geschw[0]	30000
handbetrieb.tipp.geschw[1]	30000
handbetrieb.tipp.geschw[2]	40000
handbetrieb.tipp.vb_eilgang	40000
#	
handbetrieb.jog.weg[0]	1000
handbetrieb.jog.weg[1]	2000
handbetrieb.jog.weg[2]	3000
handbetrieb.jog.geschw[0]	30000
handbetrieb.jog.geschw[1]	30000
handbetrieb.jog.geschw[2]	60000

In addition, an individual parameter value can be pre-assigned via the PLC interface. This parameter value is the first element (index 0) stored in the table of default parameter. It can be selected when an axis is activated.

Parameters can be changed at any time but they only become effective when an axis is activated (see above). When an axis is activated, the number (index) of the desired parameter set is specified in addition to the operation mode and the control element.

9.2.2.1 Continuous jog mode (by button press)

9.2.2.1.1 Control Unit

Parameterising continuous jog mode in manual mode	
Description	Control unit to manage data for parameterising continuous jog mode in manual mode, including flow control of user data.
Data type	HBTipParameterControlUnit, see description Control unit with usage check
Special features	The manual mode parameter can be written at any time and it is stored internally in a table under the index 0. This value is only becomes effective in the corresponding operation mode when the axis is activated.
Access	PLC reads HBTipParameter_Request and writes HBTipParameter_Command + X_Enable
ST path	pMC[channel_idx]^^.addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter
Commanded, requested data	
ST element	.HBTipParameter_Command .HBTipParameter_Request
Data type	HBTipParameter [► 169], for description see Section User data [► 169]
Flow control commanded data	
ST element	.X_CommandSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_RequestSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST path	.X_Enable

9.2.2.1.2 User data

Axis number	
Description	Unique system-wide number of a logical axis. The specified logical axis is assigned the velocity at which it will be moved in manual mode in continuous jog mode.
ST path	Commanded value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Command.W_LogAchsNr</code> Requested value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Request.W_LogAchsNr</code>
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Continuous joy velocity	
Description	Velocity assigned to the logical axis in continuous jog mode.
ST path	Commanded value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Command.D_Speed</code> Requested value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBTipParameterControlUnit_TipParameter.HBTipParameter_Request.D_Speed</code>
Unit	1 µm/s
Data type	UDINT

9.2.2.2 Incremental jog mode (by button press)

9.2.2.2.1 Control Unit

Programming incremental jog mode in manual mode	
Description	Control unit to manage data to parameterise incremental jog mode in manual mode, including flow control of user data.
Data type	HBJogParameterControlUnit, see description Control unit with usage check
Special features	The manual mode parameter can be written at any time and it is stored internally in a table under the index 0. This value is only becomes effective in the corresponding operation mode when the axis is activated.
ST path	PMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter
Access	PLC reads HBJogParameter_Request and writes HBJogParameter_Command + X_Enable
Commanded, requested data	
ST element	.HBJogParameter_Command .HBJogParameter_Request
Data type	HBJogParameter [► 171], for description see Section User data [► 171]
Flow control commanded data	
ST element	HBJogParameterControlUnit_JogParameter. X_CommandSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_RequestSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	.X_Enable

9.2.2.2.2 User data

Axis number	
Description	Unique system-wide number of a logical axis. The specified logical axis is assigned the velocity and incremental step width for each button press to move the axis in manual mode in incremental jog mode.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Command.W_LogAchsNr Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Request.W_LogAchsNr
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Incremental job velocity	
Description	Velocity to be assigned to the logical axis in incremental jog mode.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Command.D_Speed Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Request.D_Speed
Unit	1 µm/s
Data type	UDINT

Incremental jog path	
Description	Path traversed by the logical axis in incremental jog mode each time the incremental jog button is pressed.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Command.D_Distance Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBJogParameterControlUnit_JogParameter.HBJogParameter_Request.D_Distance
Unit	0,1 µm
Data type	UDINT

9.2.2.3 Handwheel mode

9.2.2.3.1 Control Unit

Parameterising handwheel mode in manual mode	
Description	Control unit to manage data for parameterising handwheel mode in manual mode, including flow control of user data.
Data type	HBHRParameterControlUnit, see description Control unit with usage check
Special features	The manual mode parameter can be written at any time and it is stored internally in a table under the index 0. This value is only becomes effective in the corresponding operation mode when the axis is activated.
ST path	pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.
Access	PLC reads HBHRParameter_Request and writes HBHRParameter_Command + X_Enable
Commanded, requested data	
ST element	.HBHRParameter_Command .HBHRParameter_Request
Data type	HBHRParameter [► 173], for description see Section User data [► 173]
Flow control commanded data	
ST element	.X_CommandSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_RequestSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	.X_Enable

9.2.2.3.2 User data

Axis number	
Description	Unique system-wide number of a logical axis. The specified logical axis is assigned the handwheel resolution which is the basis for moving the axis in manual mode handwheel mode.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Command.W_LogAchsNr Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Request.W_LogAchsNr
Data type	UINT
Value range	TwinCAT PLC typically [1, gNrAx]

Handwheel resolution	
Description	Resolution of axis motion path for one handwheel revolution. The internally used total resolution of the axis in 0.1 µm per applied handwheel increment results from the current handwheel resolution in 0.1 µm/increment divided by the physical handwheel resolution in increment/revolution of the handwheel specified. Manual parameter list: <pre>hr_data[0].hr_auf1_z 1000 # Incr./rev. - denominator hr_data[0].hr_auf1_n 14 # Incr./rev. - denominator</pre> Programming command (specified here in mm/revolution): #HANDWHEEL [AX=X RES1=0.1 RES2=90.2 RES3=0.5] or #SET HR [0.1, 90.2, 0.5] X (old tool)
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Command.D_Resolution Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBHRParameterControlUnit_HRParameter.HBHRParameter_Request.D_Resolution
Unit	0,1 µm / Handradumdrehung
Data type	DINT

9.2.3 Control elements for manual mode

9.2.3.1 Enforcing a button press

The HLI has four similar control units to command button presses in parallel.

This section names the control units with HBKeyControlX where the placeholder X can assume the value "" (nothing), "2", "3" through to "9". The ST Path table section lists the names of all control units present.

Transfer of GUI requests:

To transfer a request received from the GUI to the PLC, the activated control units are checked in the order HBKeyControlUnit_Key, HBKeyControlUnit_Key2, HBKeyControlUnit_Key3 through to HBKeyControlUnit_Key9 whether the request semaphore is free and the first free request element is used to transfer the button press request to the PLC. The PLC must therefore process X_RequestSemaphore for all control units for which they set X_Enable to TRUE.

9.2.3.1.1 Control Unit

Enforcing a button press in manual mode	
Description	Control unit to manage data to enforce a button press in manual mode, including flow control of user data.
Data type	HBKeyControlUnit_Key, see description Control unit with usage check
Special features	The signalling of a button press is possible at any time but it remains without any effect if there is no axis assigned in continuous/incremental jog mode.
ST path	<code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key2</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key3</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key4</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key5</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key6</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key7</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key8</code> <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_Key9</code>
Access	PLC reads HBKey_Request and writes HBKey_Command + X_Enable
Commanded, requested data	
ST element	.HBKey_Command .HBKey_Request
Data type	HBKey [▶ 177], for description see Section User data [▶ 177]
Flow control commanded data	
ST element	.X_CommandSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_RequestSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	.X_Enable

9.2.3.1.2 User data

Button number	
Description	Logical button number from which the command comes.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.W_LogKeyNr Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.W_LogKeyNr
Data type	UINT
Value range	One of the values which are defined as logical jog button pair numbers in the configuration list hand_mds.lis for the characteristics tasten_data[X].log_tasten_nr.

Start/end of button press		
Description	Start/end of button press event and direction of the motion direction keys in manual mode	
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.W_Direction Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.W_Direction	
Data type	INT	
Value range	Value	Meaning
	-1	Start of button press, negative motion direction
	0	End of button press
	1	Start of button press, positive motion direction
Special features	Consumption data item Since the enforcement of a button press is a message-orientated transfer, both the “start of button press” event and the “end of button press” event must be generated by the PLC.	

Lifetime of the button signal	
Description	If this element has a value unequal to 0, the CNC independently generates the “End of button press” event after receiving a “Start of button press” event after the time period expires and which was defined by the number of specified interpolator cycles.
ST path	Commanded value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.D_LifeTime</code> Requested value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.D_LifeTime</code>
Unit	Number of interpolator cycles
Data type	UDINT
Special features	See Note [▶ 177]

Retriggering “start of button press” event	
Description	Retriggering the “start of button press” event. If the element “Life time of a button signal” [▶ 177] has a value unequal to 0, the “start of button press” event is retriggered if the “Lifetime of the button signal” has not yet expired.
ST path	Commanded value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Command.X_FRefresh</code> Requested value <code>pMC[channel_idx]^addr^.MCControlHB_Data.HBKeyControlUnit_KeyX.HBKey_Request.X_FRefresh</code>
Data type	BOOL
Value range	[TRUE = Retriggering the “start of button press” event, FALSE]
Special features	See Note [▶ 177]

Note: The “Lifetime of the button signal” and “Retriggering the ‘start of button press’ event” elements represent a sort of watchdog function. Their use is indicated in particular if the time response of the PLC is not ensured (deterministic, e.g. soft PLC as Windows task).

9.2.3.2 Rapid traverse velocity during path motion

During continuous jog mode it is possible to switch between the normal velocity and rapid traverse velocity. Rapid traverse velocity is defined in the axis-specific parameter list.

handbetrieb.tipp.vb_eilgang (P-AXIS-00210) 4000

Here the rapid traverse is a button-specific feature and only becomes effective when the corresponding button is pushed and linked to an axis.

9.2.3.2.1 Control Unit

Rapid traverse velocity during path motion in manual mode	
Description	Control unit to activate/deactivate rapid traverse mode by a normal button press in manual mode.
Data type	HBRapidKeyControlUnit, see description Control unit with usage check
Special features	Button-specific signalling of rapid traverse mode is possible at any time and is saved internally for each button. But this only has an effect when the corresponding button is pressed.
ST path	pMC[channel_idx]^^.addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey
Access	PLC reads HBRapidKey_Request and writes HBRapidKey_Command + X_Enable
Commanded, requested data	
ST element	.HBRapidKey_Command .HBRapidKey_Request
Data type	HBRapidKey [► 179], for description see Section User data [► 179]
Flow control commanded data	
ST element	.X_CommandSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC accepts the commanded data if this element has the value TRUE and sets this element to the value FALSE after complete acceptance of the data. PLC can write data for commanding if this element has the value FALSE. The PLC sets this element to the value TRUE if all data to be commanded is written.
Flow control of requested data	
ST element	.X_RequestSemaphor
Data type	BOOL
Special features	Consumption data item
Access	CNC writes the data requested by the GUI if this element is FALSE and then sets this element to TRUE. PLC reads the data requested by the GUI if this value is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST element	.X_Enable

9.2.3.2.2 User data

Button number	
Description	Logical button number for which the rapid traverse mode should be selected/deselected.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Command.W_LogKeyNr Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Request.W_LogKeyNr
Data type	UINT
Value range	One of the values defined as logical jog button pair numbers in the configuration list hand_mds.lis for the characteristics tasten_data[X].log_tasten_nr

Rapid traverse velocity during path motion	
Description	Rapid traverse mode of the button on/off.
ST path	Commanded value pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Command.W_KeyPressed Requested value pMC[channel_idx]^addr^.MCControlHB_Data.HBRapidKeyControlUnit_RapidKey.HBRapidKey_Request.W_KeyPressed
Data type	UINT
Value range	TRUE = Button for rapid traverse mode is active. The parameterised rapid traverse path motion is used for continuous jog mode. FALSE = Button not active in rapid traverse mode. The parameterised normal path velocity is used in continuous jog mode.]

9.2.3.3 Handwheel increments

Handwheel increments, counter state	
Description	Array of control units to manage the counts of handwheel increments for all handwheels, including flow control of user data.
Data type	MCCControlSGN32Unit, see description of Control Unit
Special features	Handwheel counts can be changed at any time but they only have an effect if there is an assigned axis in handwheel mode. Relative changes to a handwheel count are only evaluated after manual mode is activated.
Access	PLC reads Request + State and writes Command + Enable
ST path	pMC[channel_idx]^addr^.MCCControlHB_Data.AMCCControlSGN32Unit_HandWheelIncs[idx]
Commanded, requested and return value	
ST element	.D_Command .D_Request .D_State
Data type	DINT
Redirection	
ST path	.X_Enable

10 Safety system

10.1 Channel-specific interface

10.1.1 Watchdog mechanism

Watchdog, CNC monitors PLC

Description	The CNC uses the data of this unit to monitor whether the PLC is still ready for operation.
-------------	---



Notice

When the CNC monitors the PLC, the PLC task in which the PLC life sign `X_alive_rw [► 182]` is set must be executed with a higher priority than the two CNC tasks SDA and COM.

In order to prevent problems with CNC block supply in large PLC projects, it is also recommended to relocate the watchdog functionality to an extra PLC task and to only assign this task with a higher priority.

Data type	MCWatchDogUnit
ST Path	pMC[channel_idx]^addr^.HLIHeadChannel_Data. MCWatchDogUnit_mc
PLC operates functionality	
Description	By setting this element to the value TRUE, the PLC signals to the CNC that it supplies the data for the watchdog functionality.
ST Element	.X_enable_w
Data type	BOOL
Value range	[TRUE = PLC supplies data for a watchdog monitor, FALSE]
Access	PLC writes
PLC signal	
Description	CNC uses this element to determine whether the PLC is still ready for operation. This check is executed when the elements “PLC operates functionality” [► 182] and “PLC present” have the value TRUE.
ST element	.X_alive_rw
Data type	BOOL
Value range	[TRUE = PLC signals that it is ready for operation, FALSE]
Access	CNC detects whether this element changes from FALSE to TRUE within the time specified by the element “Watchdog cycle time” [► 182]. The CNC sets the value to FALSE after detecting this value. CNC writes the value TRUE in this element in every CNC cycle in order to confirm its readiness for operation.
Signal “PLC not ready for operation”	
Description	If the CNC detects that the PLC is no longer ready for operation, the CNC sets this element to TRUE.
ST element	.X_alive_state_r
Data type	BOOL
Value range	[TRUE = PLC ready for operation, FALSE = PLC is not ready for operation]
Access	CNC writes
Watchdog cycle time	
Description	Cycle time of the watchdog. The values for the watchdog cycle time must be greater than the PLC cycle time in order to allow practical monitoring of the PLC by the CNC.
ST element	.D_cycle_time_w
Unit	1 µs
Data type	UDINT
Access	PLC writes

Watchdog, PLC monitors CNC

Description	The PLC uses the data of this unit to monitor whether the CNC is still ready for operation.
-------------	---



Notice

When the PLC monitors the CNC, the PLC task in which a check is made for the CNC life sign `X_alive_rw` [▶ 182] must be executed at a higher priority than the two CNC tasks `SDA` and `COM` (see the example TwinCAT [▶ 182]). In order to prevent problems with CNC block supply in large PLC projects, it is also recommended to relocate the watchdog functionality to an extra PLC task and to only assign this task with a higher priority.

Data type	LCWatchDogUnit
ST Path	<code>pMC[channel_idx]^addr^.HLIHeadChannel_Data.LCWatchDogUnit_lc</code>
CNC signal	
Description	The CNC writes TRUE to this element in every interpolation cycle in order to confirm that the CNC is ready to operation.
ST Path	<code>.X_alive_rw</code>
Data type	BOOL
Access	The CNC writes the value TRUE in every CNC cycle to confirm its readiness for operation. PLC detects whether this element changes from FALSE to TRUE within the time specified by the element "Watchdog cycle time" [▶ 182]. The PLC sets the value to FALSE after successful detection.
Signal "CNC not ready for operation"	
Description	If the PLC detects that the CNC is no longer ready for operation, it sets this element to TRUE.
ST Path	<code>.X_alive_state_w</code>
Data type	BOOL
Value range	[TRUE = CNC ready for operation, FALSE = CNC is not ready for operation]
Access	PLC writes
Watchdog cycle time	
Description	The NC kernel writes the interpolator cycle time in this element. The values for the watchdog cycle time must be greater than the PLC cycle time to permit practical monitoring of the CNC by the PLC.
ST Path	<code>.D_cycle_time_r</code>
Unit	1 µs
Data type	UDINT
Access	PLC reads

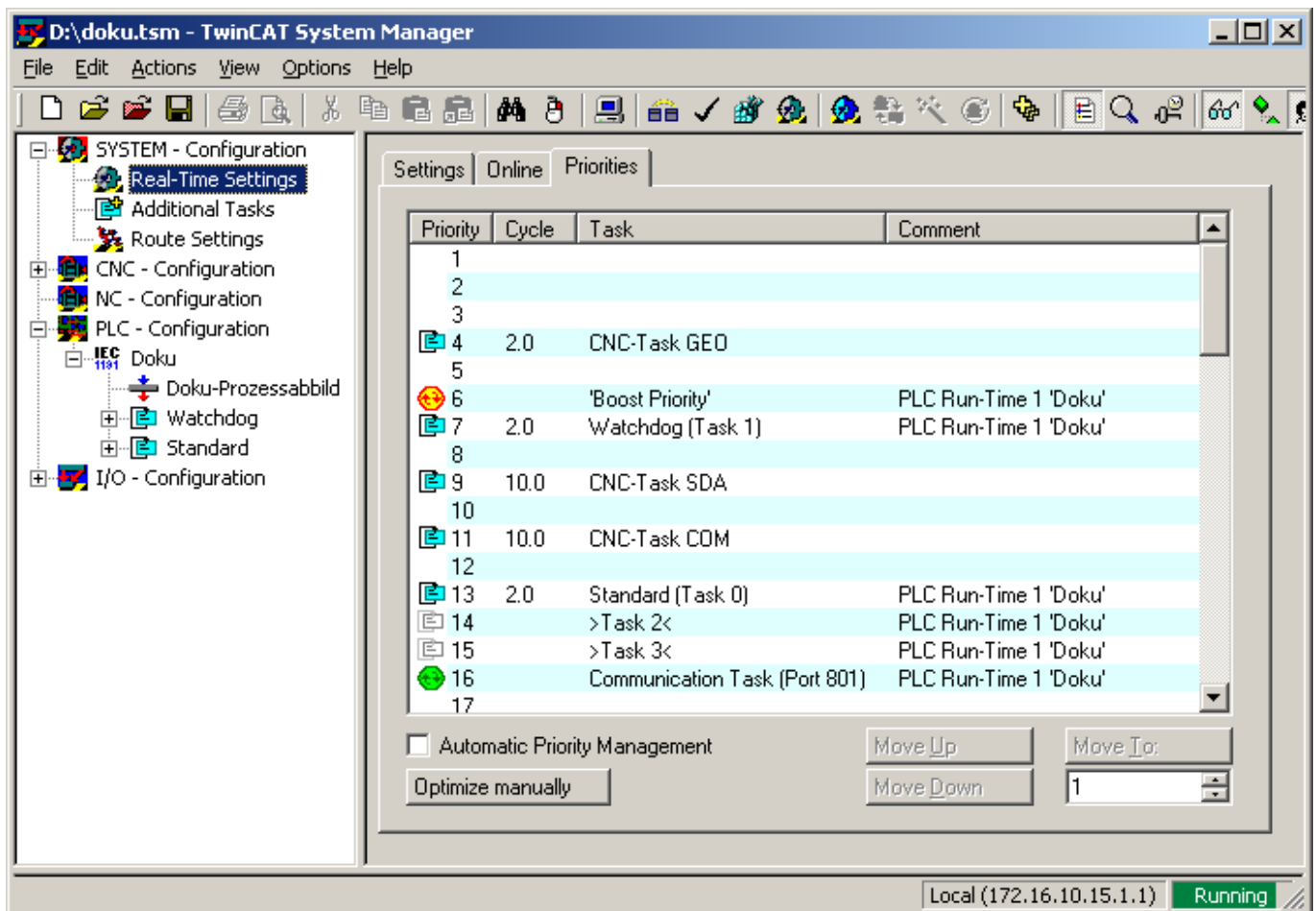


Fig. 22: Priorities for watchdog mechanism (example TwinCAT 2)

11 Management

11.1 Channel-specific interface

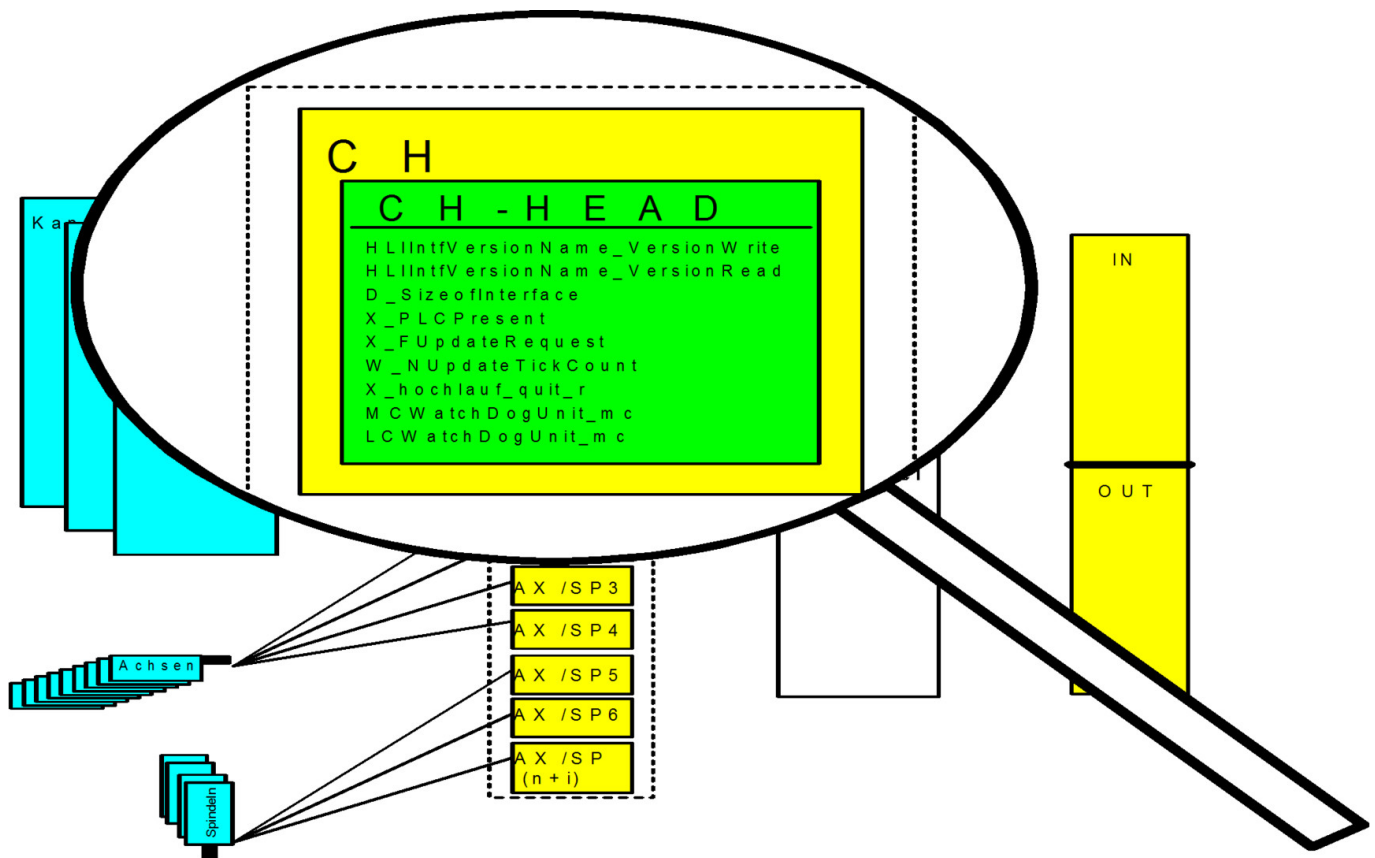


Fig. 23: Management data of the channel-specific interface

Version identifier PLC → CNC	
Description	If the version of the HLI at the NC kernel end is evaluated, the PLC stores the agreed version identifier in this element. Evaluation at the NC kernel end must be agreed for a specific application.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.HLIIntfVersionName_Version-Write.AB_Zeichen
Data type	ARRAY [1..GCW_250_HLIIntfVersionNameLength] OF BYTE
Access	PLC writes
Special features	Is currently not evaluated in the NC kernel.

Version identifier CNC → PLC	
Description	If the version of the HLI is evaluated at the PLC end, the NC kernel stores the agreed version identifier in this element. Evaluation at the PLC end must be agreed for a specific application.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.HLIIntfVersionName_Version-Read.AB_Zeichen
Data type	ARRAY [1..GCW_250_HLIIntfVersionNameLength] OF BYTE
Access	PLC reads

Size of the HLI	
Description	The NC kernel writes the size of the entire channel-specific interface area in this element.
Unit	Byte
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.D_SizeofInterface
Data type	UDINT
Access	PLC reads

PLC present	
Description	The PLC signals that it is present by setting the value to TRUE.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.X_PLCPresent
Data type	BOOL
Value range	[TRUE = PLC present, FALSE]
Access	PLC writes



Notice

Only if this value is set to TRUE can the PLC influence the NC kernel by channel-specific control units and be supplied with technology commands by the NC kernel.

NC kernel start-up completed	
Description	By setting this value to TRUE, the NC kernel signals to the PLC that the NC kernel start-up is completed and that the HLI is now supplied cyclically. This means that the display data is valid and the control commands are transferred to the NC kernel.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.X_hochlauf_quit_r
Data type	BOOL
Value range	[TRUE = Start-up of NC kernel completed, FALSE]
Access	PLC reads

Updating the HLI	
Description	By setting this value to TRUE, the PLC can initiate an update of the entire axis-specific interface area. This value is reset to FALSE by the NC kernel after the update is completed.
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.X_FUpdateRequest
Data type	BOOL
Value range	[TRUE = Updating of the HLI activated, FALSE]
Access	PLC writes/reads
Special features	Currently not supported, i.e. the HLI is updated in every NC cycle.

Update cycle of the HLI	
Description	Using this element, the PLC informs the NC kernel of the number of interrupt cycles the NC kernel may distribute the updating of the axis-specific interface area. This leads to a reduced load on the interrupt task in particular with multi-channel and multi-axis configurations.
Unit	Number of cycles of the NC kernel
ST path	pMC[channel_idx]^addr^.HLIHeadChannel_Data.W_NUpdateTickCount
Data type	UINT
Access	PLC writes
Special features	Currently not supported, i.e. the HLI is updated in every NC cycle.

11.2 Axis-specific interface

PLC present	
Description	The PLC signals that it is present by setting the value to TRUE.
Signal flow	PLC → CNC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.X_PLCPresent
Data type	BOOL
Value range	[TRUE = PLC present, FALSE]
Access	PLC writes



Notice

Only if this value is set to TRUE can the PLC influence the NC kernel by axis-specific control units and be supplied with technology commands by the NC kernel.

Logical axis number	
Description	Logical number of an axis which is unique within the entire system and identifies the axis. This datum is available after controller start-up irrespective of the configured axis mode or the assignment of the axis to a channel.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.W_LogAxNr
Data type	UINT
Access	PLC reads

Axis error	
Description	The CNC sets this value to TRUE if the axis is in error state.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^addr^.HLIHeadAxe_Data.X_Error
Data type	BOOL
Value range	[TRUE = Axis is in error state, FALSE = No error]
Access	PLC reads

Axis error, not resettable	
Description	The CNC sets this value to TRUE if an unresettable error was issued for an axis.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.HLIHeadAxe_Data.X_ResetErrorLocked
Data type	BOOL
Value range	[TRUE = Not resettable axis error occurred, FALSE = No error or resettable error occurred]
Access	PLC reads

Updating the HLI	
Description	By setting this value to TRUE, the PLC can initiate an update of the entire axis-specific interface area. This value is reset to FALSE by the NC kernel after the update is completed.
Signal flow	CNC → PLC
ST path	pAC[axis_idx]^^.addr^.HLIHeadAxe_Data.X_FUupdateRequest
Data type	BOOL
Value range	[TRUE = Updating of the HLI activated, FALSE]
Access	PLC writes/reads
Special features	Still currently supported, i.e. the HLI is updated in every NC cycle.

Update cycle of the HLI	
Description	Using this element, the PLC informs the NC kernel of the number of interrupt cycles the NC kernel may distribute the updating of the axis-specific interface area. This leads to a reduced load on the interrupt task in particular with multi-channel and multi-axis configurations.
Signal flow	PLC → CNC
Unit	Number of cycles of the NC kernel
ST path	pAC[axis_idx]^^.addr^.HLIHeadAxe_Data.W_NUupdateTickCount
Data type	UINT
Access	PLC writes
Special features	Currently not supported, i.e. the HLI is updated in every NC cycle.

12 Error messages

Errors occurring in the NC kernel can be displayed on the HLI interface in order to provide the PLC with the option of responding to the error and logging it.

The significance of the individual structure elements is documented in [DIAG].

Each error message is identified by a unique error number and the error messages are documented in [DIAG].

12.1 Management data of an error message

12.2 User data of an error message

Error number	
Description	Unique error number.
ST path	pMC[channel_idx]^addr^.MCErrSatz_Data.SatzR.Kopf. ErrorId
Data type	UDINT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Module name of the module signalling an error	
Description	Module name of the module signalling an error
ST path	pMC[channel_idx]^addr^.MCErrSatz_Data.SatzR.Kopf. ModulName
Data type	STRING(GCW_250_HLIFileNameLaengeMinus1)
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Line number	
Description	Line number in the module on which the error occurred.
ST path	pMC[channel_idx]^addr^.MCErrSatz_Data.SatzR.Kopf. Line
Data type	INT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Error number of a utility function	
Description	Error number when a utility function is used.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.UtilErrorId
Data type	UDINT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Module name of the module with utility functions signalling an error	
Description	Module name of the module with utility functions signalling an error
Signal flow	CNC → PLC
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.UtilModulName
Data type	STRING(GCW_250_HLIFileNameLaengeMinus1)
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Line number of a utility function	
Description	Line number of the line in which the error occurred in a module with utility function.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.UtilLine
Data type	INT
Special features	Used for internal diagnostic purposes.

Multiple error number	
Description	Error messages may be issued at several different points in the NC kernel. A unique multiple error number is issued to distinguish multiple usage.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf.MultipleId
Data type	UINT
Special features	Used for internal diagnostic purposes.

Type of commandable function	
Description	Type of commandable function in which an error occurred.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. BFType
Data type	UINT
Special features	Used for internal diagnostic purposes.

Channel number	
Description	Channel number of the channel in which the signalled error occurred.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. CncChannel
Data type	UINT

Communication ID	
Description	Communication ID of the BF signalling an error in the CNC
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. Kommuld
Data type	UINT
Access	PLC reads
Special features	Used for internal diagnostic purposes.

Time specification on output of an error message:	
Description	Date of the instant of the error message
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. FbZeitangabe.DateCounter
Data type	UDINT
Special features	Currently not implemented, value is always 0.

Time at error message: interrupt cycles since system start	
Description	Number of interrupt cycles since system start at the instant of an error message
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. FbZeitangabe.ZyklCounter
Data type	UDINT
Access	PLC reads

CNC version name	
Description	Version name of the CNC specified in the error message.
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. VersionName
Data type	STRING(GCW_250_HLIIntfVersNameLaengeMin1)

Recovery class	
Description	Recovery class of an error
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. BehebungsKlasse
Data type	UINT
Value range	[0, 8]

Reaction class of an error	
Description	Reaction class of an error
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. ReaktionsKlasse
Data type	UINT
Value range	[0, 8]

Body type of an error		
Description	Body type of an error. Depending on the error type, the error set body contains further information on the error which occurred. The individual structure elements are described in [DIAG].	
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Kopf. RumpfTyp	
Value range	Value	Constant
	1	HLL_RumpfTypNcProg
	2	HLL_RumpfTypMds
	3	HLL_RumpfTypKommu
	4	HLL_RumpfTypRamDisk
	5	HLL_RumpfTypFile
	6	HLL_RumpfTypIntprFile
	7	HLL_RumpfTypListeBinaer
	8	HLL_RumpfTypGcm
	9	HLL_RumpfTypLeer
	10	HLL_RumpfTypHLL
	11	HLL_RumpfTypNCPProgLr

12.2.1 Error message content, body nc program

NC program	
Description	Error information in relation to the NC program
Data type	RumpfNcProg
ST path	pNcProgErr : POINTER TO RumpfNcProg; pNcProgErr := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	Logical path number (see start-up list).
ST path	.logPfadNr
Data type	UINT
Value range	s. Start-up list
Description	Program name
ST path	.ProgName
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	Data name
ST path	.FileName
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	File offset in bytes
ST path	.FileOffset
Data type	UDINT
Description	Token offset in current NC line
ST path	.TokenOffset
Data type	UINT
Description	Block number of current NC line
ST path	.SatzNumber
Data type	UDINT
Description	Additional program information
ST path	.SaddProgInfo
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)

12.2.2 Error message content, body machine data

Machine data	
Description	Error information in relation to an update of machine data
Data type	RumpfMds
ST path	pMachineDataErr : POINTER TO RumpfMds; pMachineDataErr := ADR(pMC[channel_idx]^ .addr^ .MCError-Satz_Data.SatzR.Rumpf.Maske);
Description	Type of list
ST path	.ListenTyp
Data type	UINT
Value range	1 - manual data, 2 – channel list, 3 – axis list, 4 – zero offsets, 5 – tool data, 6 – clamp position, 7 – start-up list, 9 – axis compensations, 12 – external variables
Description	Name of wrong structure
ST path	.StruktName
Data type	STRING(GCW_250_HLIStruktNameLaengeMinus1)

12.2.3 Error message content, body communication

Communication	
Description	Error information in relation to a protocol data unit (message)
Data type	RumpfKommu
ST path	pCommuErr : POINTER TO RumpfKommu; pCommuErr := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	Type of list
ST path	.Medium
Data type	UINT
Value range	1 – function block, 2 – PDU
Description	Code of message
ST path	.typ
Data type	UDINT
Description	Receiver or sender of message
ST path	.Partner
Data type	UDINT

12.2.4 Error message content, body RAM disk

RAM disk	
Description	Error information in relation to RAM disk access
Data type	RumpfRAMDisk
ST path	pRamDiskErr : POINTER TO RumpfRamDisk; pRamDiskErr := ADR(pMC[channel_idx]^^.addr^.MCErr- Satz_Data.SatzR.Rumpf.Maske);
Description	Listentyp
ST path	.Medium
Data type	UINT
Value range	1 – function block, 2 – PDU
Description	Code of message
ST path	.typ
Data type	UDINT
Value range	
Description	Receiver or sender of message
ST path	.Partner
Data type	UDINT
Value range	
Description	File name
ST path	.FileName
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	Dateoffset in Bytes
ST path	.FileOffset
Data type	UDINT

12.2.5 Error message content, body file

File	
Description	Error information in relation to file access
Data type	RumpfFile
ST path	pFileErr : POINTER TO RumpfFile; pFileErr := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	File name
ST path	.FileName
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	File offset in bytes
ST path	.FileOffset
Data type	UDINT

12.2.6 Error message content, body interpreted file list

List interpretation		
Description	Error information in relation to interpretation of ASCII parameter list	
Data type	RumpfIntprFile	
ST path	pIntprFileErr: POINTER TO RumpfIntprFile; pIntprFileErr := ADR(pMC[channel_idx]^ .addr^ .MCErr- Satz_Data.SatzR.Rumpf.Maske);	
Description	File name	
Data type	.FileName	
ST path	STRING(GCW_250_HLIProgNameLaengeMinus1)	
Description	File offset in bytes	
Data type	.FileOffset	
ST path	UDINT	
Description	Line number in current file	
Data type	.FileLine	
ST path	UDINT	
Description	Listentyp	
Data type	.ListenTyp	
ST path	UINT	
Value range	Value	Meaning
	1	Manual mode list
	2	Channel parameter list
	3	Axis parameter list
	4	Zero offset list
	5	Tool parameter list
	6	Position offset list
	7	Start-up list
	9	Axis compensation list
Description	Name of wrong structure	
Data type	.StruktName	
ST path	STRING(GCW_250_HLIStruktNameLaengeMinus1)	

12.2.7 Error message content, body binary list

Binary list	
Description	Error information in relation to binary list update
Data type	pBinaryListErr : POINTER TO RumpflntprFile; pBinaryListErr := ADR(pMC[channel_idx]^ .addr^ .MCErr- Satz_Data.SatzR.Rumpf.Maske);
ST path	pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske. NCProg
Description	Name of wrong structure
ST path	.StruktName
Data type	STRING(GCW_250_HLIStruktNameLaengeMinus1)

12.2.8 Error message content, body global channel manager

Global channel manager	
Description	Error information in relation to Global Channel Manager
Data type	RumpfListeGcm
ST path	pGcmErr : POINTER TO RumpfGcm; pGcmErr := ADR(pMC[channel_idx]^ .addr^ .MCErrSatz_Data.SatzR.Rumpf.Maske);
Description	Name of wrong structure
ST path	.Token
Data type	STRING(GCW_250_HLIVarStringLaengeMinus1)
Description	File name
ST path	.FileName
Data type	STRING(GCW_250_HLIProgNameLaengeMinus1)
Description	Interpreter number
ST path	.InterpNo
Data type	UDINT
Description	Line number
ST path	.LineNo
Data type	UDINT
Description	Column number
ST path	.ColNo
Data type	UDINT
Description	Offset
ST path	.OffsetNo
Data type	UDINT
Description	Number of command
ST path	.CommandNo

Data type	UDINT
Description	Part ID
ST path	.PartId
Data type	UDINT
Description	Step ID
ST path	.StepId
Data type	UDINT
Description	Level of if constructs
ST path	.IfLevel
Data type	UDINT

12.2.9 Additional error information value 1 - 5

Additional error information	
Description	Individual error information
ST path	pMC[channel_idx]^^.addr^.MCErrSatz_Data.SatzR.Rumpf. Wert1 ... Wert5
Data type	WertB
Description	Data type
ST path	.typ
Data type	INT
Value range	See Value range of variable 'Type' of additional error information [► 206]
Description	Dimension of datum
ST path	.Dimension
Data type	INT
Value range	See Value range of 'Dimension' variable of the additional error information [► 207].
Description	Significance of datum
ST path	.bedeutung
Data type	INT
Value range	See Value range of 'Significance' variable of additional error information [► 208].
Description	Datum itself
ST path	.Contents
Data type	(as union, must be type-caste corresponding to the specified data type)

12.2.9.1 Value range of variable 'Type' of additional error information

Constant	Value	Description
HLI_TYP_BOOLEAN	1	Boolean value
HLI_TYP_UNSO8	2	Unsigned 1 byte value
HLI_TYP_SGN08	3	Signed 1 byte value
HLI_TYP_UNSO16	4	Unsigned 2 byte value
HLI_TYP_SGN16	5	Signed 2 byte value
HLI_TYP_UNSO32	6	Unsigned 4 byte value
HLI_TYP_SGN32	7	Signed 4 byte value
HLI_TYP_UNSO64	8	Unsigned 8 byte value
HLI_TYP_SGN64	9	Signed 8 byte value
HLI_TYP_REAL64	10	8 byte decimal value*
HLI_TYP_POINTER	12	Address
HLI_TYP_CHAR	18	Character, 1 byte
HLI_TYP_STRING	19	Character string
HLI_TYP_ADRESSE	20	Address
HLI_TYP_A3_REAL64	23	Array containing three 8-byte decimal values
HLI_TYP_HIGH_RES_SGN_POS	24	Signed value, size depends on system and may be 4 or 8 bytes.
HLI_TYP_BITARRAY_32	25	Array of 4-byte values
HLI_TYP_BITARRAY_16	26	Array of 2-byte values

[Back to description of 'Type' variable.](#)

12.2.9.2 Value range of 'Dimension' variable of the additional error information

Constant	Value	Unit	Description
HLI_NO_DIMENSION	-1	-	This is not a dimension specification
HLI_DIM_DIMENSIONSLOS	0	-	Without dimension
HLI_DIM_POSITION	1	10^{-4} mm or 10^{-4} °	Position
HLI_DIM_GESCHWINDIGKEIT	2	10^{-3} mm/s or 10^{-3} °	Velocity
HLI_DIM_BESCHLEUNIGUNG	3	1 mm/s^2 or $1^\circ/\text{s}^2$	acceleration
HLI_DIM_RUCK	4	1 mm/s^3 or $1^\circ/\text{s}^3$	jerk
HLI_DIM_ZEIT	5	1 µs	time
HLI_DIM_PROZENT	6	0,1 %	Prozent
HLI_DIM_ADRESSE	7	-	Address
HLI_DIM_INKREMENTE	8	Increments	Path increment
HLI_DIM_UMDR_VORSCHUB	9	10^{-4} mm/rev	Feedrate per revolution
HLI_DIM_V_SCHNITT	10	10^{-3} mm/s	Cutting velocity
HLI_DIM_WEG_AUFLOESUNG	11	Increments/ 10^{-3} mm	Path resolution
HLI_DIM_INKR_UMDREHUNG	12	Increments/rev.	Increments per revolution
HLI_DIM_BYTE	13	-	Number of bytes
HLI_DIM_PROPORTIONAL_GAIN	14	0.01/s	Proportional gain
HLI_DIM_FREQUENCY	15	Hz	Frequency
HLI_DIM_LOAD	16	1 kg or $1 \text{ kg}\cdot\text{m}^2$	Motor load
HLI_DIM_POSITION_HIGH_RES	17	10^{-8} mm or 10^{-8} °	Position, high resolution
HLI_DIM_INKREMENTE_HIGH_RES	18	10^{-4} increments	Increment, high resolution

Back to description of variable dimension.

12.2.9.3 Value range of 'Significance' variable of additional error information

Constant	Value	Description
HLI_BEDEUT_IGNORE	0	Ignore value may not be changed: Initialise with 0
HLI_BEDEUT_GRENZ_WERT	1	Limit value
HLI_BEDEUT_AKT_WERT	2	Current value
HLI_BEDEUT_FEHL_WERT	3	Error value
HLI_BEDEUT_ERWARTET_WERT	4	Expected value
HLI_BEDEUT_KORR_WERT	5	Corrected value
HLI_BEDEUT_LOG_ACHS_NR	6	logical axis number
HLI_BEDEUT_ANTR_TYP	7	Drive type
HLI_BEDEUT_LOG_BED_ELEM_NR	8	logical identifier of control element
HLI_BEDEUT_ZUSTAND	9	State
HLI_BEDEUT_TRANSITION	10	Transition
HLI_BEDEUT_SENDER	11	Sender
HLI_BEDEUT_KLASSE	12	Class
HLI_BEDEUT_INSTANZ	13	Instance
HLI_BEDEUT_IDENT_NR	14	Identification number
HLI_BEDEUT_STATUS	15	Status
HLI_BEDEUT_RING_NR	16	Ring number
HLI_BEDEUT_SATZ_NR	17	Block number
HLI_BEDEUT_MIN_LIMIT	18	Lower limit value
HLI_BEDEUT_MAX_LIMIT	19	Upper limit value
HLI_BEDEUT_START_WERT	20	Initial value
HLI_BEDEUT_ZIEL_WERT	21	Final value
HLI_BEDEUT_FILENAME	22	File name
HLI_BEDEUT_LINE_NUMBER	24	Line number in a file
HLI_BEDEUT_COLUMN_NUMBER	25	Column number in a file
HLI_BEDEUT_ARGUMENT	26	Identifier of an argument
HLI_BEDEUT_PARAMETER	27	Identifier of a parameter
HLI_BEDEUT_AXIS	28	Identifier of an axis
HLI_BEDEUT_COMPENSATION	29	Compensation index
HLI_BEDEUT_IDENTIFIER	30	Identifier
HLI_BEDEUT_CHAIN	31	Kinematic chain

[Back to description of variable significance.](#)

13 Messages

13.1 Control Unit

Message to/from PLC	
Description	This control unit receives messages sent by the CNC to the PLC and messages sent by the PLC to other users of the communication process provided by the CNC. The CNC can send a message to the PLC via the commands #MSG PLC ["..."] or #MSG SYN PLC ["..."] (see [PROG]). The information on which of the commands was used to send a message is also sent in the user data of this control unit.
Data type	McControlMsgUnit, see description Control unit with usage check
ST path	pMC[channel_idx]^addr^ MsgMcControl_Data
Commanded, requested data	
ST element	.CommandW .RequestR
Data type	HliMsgSendung
Access	PLC writes CommandW and reads RequestR
Flow control of commanded value	
ST element	.CommandSemaphorRw
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	PLC can write data for commanding if CommandSemaphorRw has the value FALSE. When all data is written, the PLC sets CommandSemaphorRw to the value TRUE. CNC takes the commanded data if CommandSemaphorRw is TRUE and then sets the flag to FALSE.
Flow control of requested data	
ST element	.RequestSemaphorRw
Data type	BOOL
Value range	[TRUE, FALSE]
Special features	Consumption data item
Access	The data requested by the GUI can be written to the control unit if RequestSemaphorRw is FALSE. This element is then set to TRUE. PLC reads the data requested by the GUI if RequestSemaphorRw is TRUE. After the PLC fully accepts the data, the PLC sets this element to FALSE.
Redirection	
ST path	.EnableW

13.2 User data

13.2.1 Requested and commanded user data

Content of message to/from PLC	
Description	Data of message which was received or sent.
ST path	Commanded value pMC[channel_idx]^^.addr^.MsgMcControl_Data.CommandW Requested value pMC[channel_idx]^^.addr^.MsgMcControl_Data.RequestR
ST element	.Contents
Data type	ARRAY [1 .. GCW_250_HliMessageLaenge] OF BYTE

Type of message to/from PLC		
Description	Marks whether a message was sent by CNC to the PLC by a #MSG or #MSGINFO command (see @@[PROG]).	
ST path	Commanded value pMC[channel_idx]^^.addr^.MsgMcControl_Data.CommandW Requested value pMC[channel_idx]^^.addr^.MsgMcControl_Data.RequestR	
ST element	.typ	
Data type	UDINT	
Value range	Value	Meaning
	0	Initial value
	1	Type of message which was sent by a #MSG command (if it was sent by the CNC)
	2	Type of message which was sent by a #MSGINFO command (if it was sent by the CNC)

14 Implementation as PLC library

To access the HLI, a PLC library is provided for use by the PLC application developer. This library is implemented specifically for various PLC systems. The table below contains the names of the libraries for the supported PLC systems.

PLC system	KW	3S	TwinCAT
HLI PLC library	-	-	tccnchli1.lib

This library implements

- data structures which are components of the HLI.
- global variables that are used to access the elements of the HLI
- function blocks which are used to initialise global pointers

14.1 Access to HLI

14.1.1 PLC system TwinCAT

In this system, the global pointers to specific areas of the HLI are created in the HLI-PLC library. These pointers are used to access control units and status data or other data for this type of HLI area. The table below shows the uses of these pointers and in each of the individual PLC systems:

	Explanation	PLC systems
Global pointers		TwinCAT
pPlatform	Platform-specific area	X
pMC	Array of pointers to channel-specific areas	X
pAC	Array of pointers to axis-specific areas	X
pCTM	Array of pointers to job planning areas	X
pVEGlob	Pointers to global V.E variables	X
pVE	Array of pointers to channel-specific V.E variables	X

Global pointers to HLI areas

Make sure only to initialise the pointers which make sense depending on the configuration of the NC kernel (number of channels number of axes, etc.). All other pointers are ZERO pointers.

14.2 Function blocks within the PLC library

14.2.1 Overview of the PLCopen FBs

The table below provides an overview of the availability of the function blocks implemented in the PLC systems supported by ISG.

Function block	PLC systems		
	KW	3S	TwinCAT
ISG_HliInterface	-	-	X

14.2.2 ISG HLI Interface

An instance of the ISG_HliInterface must be **definitely** invoked by every PLC application for whose system this FB was implemented and which wants to access the HLI elements.

The NC kernel requests information about the HLI configuration (number of channels, number of axes, etc.) on a rising edge at the bStart input and compares the HLI configuration with the PLC configuration. If differences occur, the bError output is set to TRUE and an error message identifier is output at the iErrorID output. On the other hand, if the attributes of the HLI are identical on both sides, the bInitialized output is set to TRUE and the PLC application can access the relevant areas of the HLI via the global defined pointers (see PLC main program frame).

Block diagram

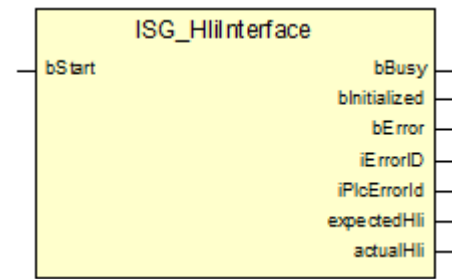


Fig. 24: FB parameters

VAR_INPUT		
Variable name	Data type	Description
bStart	BOOL	Rising edge triggers a check on whether the HLI definitions of the PLC and the NC kernel are identical.

VAR_OUTPUT		
Variable name	Data type	Description
bBusy	BOOL	Request and checking process are still active.
bInitialized	BOOL	The check was successfully completed and now access to the HLI is permitted for the first time.
bError	BOOL	Is TRUE if an error occurs in the FB.
iErrorId	UDINT	Error identifier. For possible values, see errors. Reference source not found.
iPlcErrorId	UDINT	Specific error message of the PLC system used. For further explanations, see errors. Reference source not found. Error value 9 (ISG_PLC_PFORM_ERROR)
expectedHli	HLI_DIAGNOSTIC	Information regarding the HLI configuration on the PLC side. The data structure contains the number of main areas of the HLI and the version identifier of the HLI definitions.
actualHli	HLI_DIAGNOSTIC	Contains the HLI configuration on the NC kernel side.

Behaviour of the FB:

- Table of error identifiers

• Error value	• Global constant • defined in PLC library	• Description
• 0	• ISG_NO_ERROR	• No error occurred
• 1	• ISG_WRONG_VERSION	• Version identifier of HLI from the PLC and NC kernel are different.
• 2	• ISG_WRONG_PARAMETER	• Different configuration (number of channels or number of axes or etc.) of the HLI in PLC and NC kernel.
• 9	• ISG_PLC_PFORM_ERROR	• The PLC system requested the NC kernel to send a description of the interface (HLI) on the NC side. This request was acknowledged negatively. An error value specific to the PLC system is output at the iPlcErrorId output, if available (example: with TwinCAT, this case involves an ADS error whose error number is output at the iPlcErrorId output). • • Reason for this error message: • On a TwinCAT system, an empty CNC configuration for the CNC was activated with the system manager. • • Solution: • Activate a CNC configuration which contains the axes and/or channels for the CNC.

Error value from ISG_HliInterface

15 Programming examples

15.1 PLC main program frame

```

PROGRAM MAIN
VAR
    HLI                               : ISG_HliInterface;

    HliInitError                      : BOOL := FALSE; (* Error on initialisation of
HLI *)
    UserInitialisationDone : BOOL := FALSE; (* User initialisation done *)
END_VAR

(* Request description of the HLI from the CNC *)
Hli(bStart := TRUE);

(* Check if initialisation of HLI ended successfully and if
errors occurred during initialisation phase. *)

IF Hli.bInitialized = TRUE AND Hli.bError = FALSE
THEN
    (* Do the initialization we do once the PLC starts up. *)
    IF UserInitialisationDone = FALSE THEN
        (* Get the result of the user defined initialisation *)
        UserInitialisationDone := UserInitialisations [ 214 ] (dummy:=TRUE);
    END_IF;

    (* ----- *)
    (* Insert your PLC application code after this comment *)
    (* ----- *)

    IF Hli.bError = TRUE THEN
        (* Error on initialisation of the HLI *)
        (* iErrorId contains error number *)

        HliInitError := TRUE;
    END_IF;
END_IF;

```

15.1.1 Initialisation function UserInitialisations()

```

FUNCTION UserInitialisations : BOOL

VAR_INPUT
    dummy : BOOL := FALSE; (* not_used *)
END_VAR
VAR
    AxIdx : UDINT;
    ChIdx : UDINT;
END_VAR

```

```

(* Register PLC at all axes interfaces *)
FOR AxIdx := 1 TO nAxis DO
  (* Set plc_present_w at each axis *)
  pAC[AxIdx]^addr^.HLIHeadAxe_Data.X_PLCPresent := TRUE;

  (* Register at all axis specific control units you want to handle by
  PLC *)
  (* Register at all control units to enable a drive *)
  pAC[AxIdx]^addr^.McControlLr_Data.MCControlBoolUnit_TorquePermis-
  sion.X_Enable
    := TRUE;
  pAC[Laufl]^addr^.McControlLr_Data.MCControlBoolUnit_ReleaseFeed-
  hold.X_Enable
    := TRUE;
  pAC[Laufl]^addr^.McControlLr_Data.MCControlBoolUnit_DriveOn.X_En-
  able := TRUE;

END_FOR;

(* Register PLC at all channel interfaces *)
FOR ChIdx := 1 TO nChannel DO
  (* Set plc_present_w at each channel *)
  pMC[ChIdx]^addr^.HLIHeadChannel_Data.X_PLCPresent := TRUE;

  (* Register at all channel specific control units you want to handle
  by PLC *)

END_FOR;

UserInitialisations := TRUE;

```

Keyword index

A

Antriebsdaten lesen.....	39
Antriebsdaten zyklisch lesen/schreiben	72
Antriebsdrift	41
Antriebstyp	35
AxeHBDisplayData_Data	162
Abstandsregelung	
Achse	
Achsgruppe	
Achsname	
Achsnummer	
Achstyp	
ACS	
Aktivierung von Bedienelementen	
Aktualisierung	
Antrieb	
Anzahl	
assignment	
Auflösung	
Auftragsfehler	
Ausführungsdauer	
axis	
Axis supply	

Beauftragung	66	Referenzpunktfahrt erfolgt	29
Kommando	68	Handbetrieb	165
Status	69	HLI:achsenspezifisch	190
Abstandsregelung:Beauftragung	66	HLI:kanalspezifisch	188
Abstandsregelung:Kommando	68	betriebsbereit	37
Abstandsregelung>Status	69	Daten	39
fährt vorwärts	31	EIN	48
Fehler	189	Fehler	38
Fehler:nicht resetbar	190	Istwert:gültig	32
Gantry:Differenz:ausfahren	56	Leistungszuschaltung:bereit zur	36
Identifikation	21	Zustand	38
im Regelfenster	29	Technologiefunktion:nicht quittiert:satzübergreifend	144, 146, 147
in Position	30	Technologiefunktion:satzübergreifend:Achse... ..	136
Interpolator:Zustand	33	Technologiefunktion:satzübergreifend:Kanal	137
ist bewegt	30	Technologiefunktion:satzweise:Achse	135
Konfiguration:Modus	32	Technologiefunktion:satzweise:Kanal	136
Kopplung:Beauftragung	62	axis:logical	33
Kopplung:Kommando	64	Handrad	163
Kopplung:Nummer:Master	34	Spindel	80
Kopplung>Status	63	H-Funktion	143
Kopplung:Typ	34	M-Funktion	143
Kopplung:Vorschrift	65	S-Funktion	145
Nachführbetrieb	55	assignment:logical	33
Name:PCS	21	deactivate	61
Nummer:Gantry-Master	32	park	61
Nummer:logische	22, 189	physical axis to logical axis	31
Position:ACS	25		
Position:Messwert	60		
Position:PCS	23		
Referenzposition	58		
Referenzposition:Löschen	59		
Referenzposition:Übernahme	57		
Simulation:Bearbeitungs	52		
Sollwerte:externe	21		
Spindel:ACS	22		
Technologiefunktion:Daten	138		
Typ:ACS	35		
Typ:PCS	22		
Vorschubfreigabe	45		
Vorschuboverride	47		
Vorschuboverride gültig	48		
Vorschubstopp EIN/AUS	46		
Warten auf angeforderte	95		
wird verfahren	30		
Zustand:ACS	28		
Zustand:PCS	27		
in Position	94		
Warten auf in Position	94		
PCS	21		
Handbetrieb:Aktivierung von Bedienelementen	166		
Handbetrieb:Handradbetrieb	174		
Handbetrieb:Jogbetrieb	172		
Handbetrieb:Tippbetrieb	170		
S-Funktion	146		
ACS	35		
PCS	22		
Achse:Spindel	22		
Achse:Typ	35		
Achse:Zustand	28		
Position:Ist	26		
Position:Soll	26		
Position:Ziel	25		

B

Bahngeschwindigkeit -Vorschubaktivierung externe	119
Bedienelementnummer	163
Betriebsart	73
Bahn	
Bahngeschwindigkeit -vorschub	
Bearbeitung	
Bearbeitungssimulation	
Beauftragbare Funktion	
Beauftragung	
Bedienelement	
Behebungsklasse	
Betriebsartenumschaltung	
Betriebszustand	
Bewegung	
Box	

Stopp:Grund	91
Tangente	102
Tangente:Normalenvektor	103
Tangente:Vektor	103
aktuell	90
externe Vorgabe	118
Grenzwert unterschritten	96
programmiert	89
programmiert:Echtzeiteinfluß	90
Simulation:Achse	52
EIN/AUS	115
Typ:bei Fehlermeldung	193
Spindel:extern	83
Spindel:Typ	85
Nummer	163
Handbetrieb:Aktivierung von Bedienelementen	166
bei Fehlermeldung	194
Ausgangs	155
Ausgangszustand:Betriebsartwechsel	156
Control Unit	154
Handbetrieb	162
Istwert	159
Parameter:Betriebsartwechsel	158
Zielzustand:Betriebsartwechsel	157
Zustands-Istwert	159
Zielzustand	156
Spindel	78
Bahn:Satznummer	97
fortsetzen:Kollision	97
Fortsetzung	114
Freigabe	94
TwinCAT:Ads NetId	44
TwinCAT:Kanal	44

C

Channel	
Command value	
Control Unit	
Operation mode	108
Activation:external	20
Betriebsart	154

D

DRIVE_STATE_MODE_0	39
DRIVE_STATE_MODE_1	39
DRIVE_STATE_MODE_2	40
DRIVE_STATE_MODE_3	40
DRIVE_STATE_MODE_4	40
DRIVE_STATE_MODE_5	40
DRIVE_STATE_MODE_6	41
Daten	
deactivate	
Dekodierung	
Dienstfunktion	
Drehrichtung	
Drehzahl	
Drehzahlüberwachung	
Durchsetzung eines Tastendrucks	

Antrieb	39
Technologiefunktion:Achse	138
Technologiefunktion:Kanal	140
axis	61
NC-Programm:Fortsetzung	106
NC-Programm:Stopp	107
Fehlernummer	192
Spindel	86
programmiert	77
aktiv	79
ungültig	80
Handbetrieb	176

E

Einzelsatzbetrieb	113
Ecke	
Eilgang	
Eilganggeschwindigkeit	
Eilganggeschwindigkeit während Verfahrbewegung	
Einlesefreigabe	
Einleesperre	
Einzelsatzmodus	
extern	
external	
externe	
Grenzwert:Abstand:nach	100
Grenzwert:Abstand:vor	100
Wartezeit	100
Override:Kanal	116
Bewegung	97
Handbetrieb	179
Handbetrieb:Eilganggeschwindigkeit während Verfahrbewegung	180
Unterdrückung	52
Interpolator	119
NC-Satz	105
Vorgabe Bahngeschwindigkeit	118
Command value:Activation	20
Bahngeschwindigkeit:Aktivierung	119

F

Fahrweg	84
Fehlernummer	191
Fahren	
Fehler	
Fehlermeldung	
Fehlermeldungsrumpf	
Fläche	
Fortsetzposition	
Fortsetzung	
Freigabe	
Funktionsbaustein	

rückwärts	127
aktuell:NC-Programm	89
aktuell:NC-Satz	88
Rest:NC-Satz	28
Rest:verwerfen:Ende-Marke	126
Rest:verwerfen:Kommando	125
Rest:verwerfen:Zustand	99
Achse	189
Achse:nicht resetbar	190
Behebung erwartet	93
Spindel	82
warten auf externe Vorgabe	93
Beauftragbare Funktion:Typ	193
Behebungsklasse	194
Fehlernummer	191
Kanalnummer	193
Kommunikations-Id	193
Modulname	191
Modulname:Dienstfunktion	192
Nummer:Dienstfunktion	192
Nummer:Mehrfachfehler	192
Reaktionsklasse	194
Rumpftyp	195
Versionsbezeichnung CNC	194
Zeilennummer	191
Zeilennummer:Dienstfunktion	192
Zeitangabe:Interruptzyklen	193
Zeitangabe:Interruptzyklen seit Systemstart	193
binary list	202
Datei	200
global channel manager	203
IListeninterpretation	201
Kommunikation	198
Maschinendaten	197
NC program	196
RAM disk	199
Wert 1 – 5	205
Dienstfunktion	192
Mehrfachfehler	192
Normalenvektor	103
Satzvorlauf:Abstand	96
der Bewegung	114
NC-Programm:Dekodierung	106
Bearbeitung nächster NC-Satz	106
Bewegung	94
Regler:Kommando	49
Regler:Zustand	31
ISG_HliInterface	212

G

G-Funktion	84
G01	
G100	
G108	
G200	
G201	
Gantry	
Gerät	
Gesamtoffset	
Geschwindigkeit	
Grenzwert	

Größe	
Zeitangabe	99
aktiv	99
aktiv	99
aktiv	98
aktiv	98
Differenz:ausfahren	56
Masterachse:Nummer	32
Identifikation:TwinCAT	44
PCS	25
Achse:Reduktion	50
Achse:reduziert, Zone 1	51
Achse:reduziert, Zone 2	51
Bahn:Grenzwert unterschritten	96
Eilgang:Achssbewegung	97
Kanal:Reduktion	122
Kanal:reduziert, Zone 1	122
Kanal:reduziert, Zone 2	123
minimale:Werkzeug:Ignorierung	53
NC-Satz, aktuell:Ende	101
Ecke:Abstand:nach	100
Ecke:Abstand:vor	100
HLI	187

H

Handradauflösung	163
HBActivation_Command	166
HBActivationControlUnit_Activation	165
Halt	
Handbetrieb	
Handbetriebsart	
Handbetriebsparametersatz	
Handrad	
Handradbetrieb	
Handradinkremente Zählerstand	
Hardware	
H-Funktion	
HLI	
Hochlauf	

Bahn:Bedingung	91
Nothalt:Kanal	113
Nothalt:Spindel	82
wahlweiser	114
Achsnnummer:Aktivierung von Bedienelementen	166
Achsnnummer:Handradbetrieb	174
Achsnnummer:Jogbetrieb	172
Achsnnummer:Tippbetrieb	170
aktiv:mit paralleler Interpolation	98
aktiv:ohne parallele Interpolation	98
Aktivierung von Bedienelementen	165
Bedienelementnummer:Aktivierung von Bedienelementen	166
Betriebsart	162
Durchsetzung eines Tastendrucks	176
Eilganggeschwindigkeit während Verfahrbewegung	179
Eilganggeschwindigkeit:Eilganggeschwindigkeit während Verfahrbewegung	180
Handbetriebsart:Aktivierung von Bedienelementen	167
Handradauflösung	174
Handradbetrieb	173
Handradinkremente Zählerstand	181
Jogbetrieb	171
Joggeschwindigkeit	172
Jogweg	172
Lebenszeit des Tastensignals:Durchsetzung eines Tastendrucks	178
Offset:zurück fahren:Anhalten	55
Retriggern Tastendruck-Beginn-Ereignisses:Durchsetzung eines Tastendrucks	178
Tastendruck Beginn/Ende:Durchsetzung eines Tastendrucks	177
Tastennummer:Durchsetzung eines Tastendrucks	177
Tastennummer:Eilganggeschwindigkeit während Tastendruck	180
Tippbetrieb	169
Tippgeschwindigkeit	170
Zustand	162
Aktivierung von Bedienelementen	167
Handbetrieb:Aktivierung von Bedienelementen	167
Auflösung	163
Handbetrieb	174
Achsnnummer	174
Auflösung	174
Parametrierung	173
Handbetrieb	181
extern:Messen	74
extern:Messen:Auftrag	75
extern:Messen:Flanke	75
extern:Messen:Nummer:Auftrag	75
extern:Messen:Nummer:Eingang	75
Ausführungsdauer, voraussichtlich	143
Daten	143
Nummer	143
Programmzeilennummer	144
Satznummer	143
Zusatzinformation	144
Aktualisierung:achsspezifisch	190

Aktualisierung:kanalspezifisch	188
Größe	187
Updatezyklus:achsspezifisch	190
Updatezyklus:kanalspezifisch	188
Zeiger	211
NC-Kern, abgeschlossen	188

I

ISG_HllInterface	212
Istdrehzahl	77
Index	
Inhalt	
Interpolator	
Istposition	
Platzversatzgruppe	109
Nachricht:an/von PLC	210
Achse:Zustand	33
aktiv	93
Einlesesperre	119
ACS	26
PCS	24
Spindel	78

J

Jogbetrieb	
Joggeschwindigkeit	
Jogweg	
Achsnnummer	172
Geschwindigkeit	172
Parametrierung	171
Verfahrgerwindigkeit	163
Verfahrweg	164
Weg	172
Handbetrieb	172
Handbetrieb	172

K

Kanal	
Kanalnummer	
Kollision	
Kommunikations-Id	
Kompensation	
Konfiguration	
Kopplung	

Nummer.....	23
Technologiefunktion:Daten.....	140
Vorschubstopp EIN/AUS	110
bei Fehlermeldung.....	193
Betriebsart:Umschaltung	159
Bewegung:fortsetzen.....	97
bei Fehlermeldung.....	193
Antriebsdrift	41
Spindelsteigungsfehler	41
Temperatureinfluss	42
Umkehrspanne	42
Achse:Modus.....	32
Achse:Beauftragung	62
Achse:Kommando	64
Achse:Nummer:Master.....	34
Achse:Status	63
Achse:Typ	34
Achse:Vorschrift	65

L

Lebenszeit des Tastensignals	
Handbetrieb:Durchsetzung eines Tastendrucks	
.....	178

M

Messsignal	59
Messwert gültig	43
M-Funktion	84
manual mode	
MCS	
Mehrfachfehlernummer	
Messen	
Messtaster	
Messvorgang	
Messwert	
Modulname	

offset:move back:stop.....	54
Position:Soll:Werkzeugmittelpunkt	27
bei Fehlermeldung	192
Hardware:extern	74
Hardware:extern:Auftrag	75
Hardware:extern:Flanke	75
Hardware:extern:Nummer:Auftrag	75
Hardware:extern:Nummer:Eingang	75
Zustand.....	43
aktiv	99
Position:Achse	60
Ausführungsdauer, voraussichtlich.....	143
Daten	143
Nummer	143
Nummer:Spindelpositionierfunktion	146
Nummer:Spindelschaltfunktion	145
Programmzeilennummer	144
Satznummer	143
Zusatzinformation	144
bei Fehlermeldung	191
Dienstfunktion:bei Fehlermeldung	192

N

Nachfuhrbetrieb	55
Nachricht	
NC block	
NC-Kern	
NC-Programm	
NC-Satz	
Nothalt	
Nummer	
an/von PLC.....	209
an/von PLC:Inhalt	210
an/von PLC:Typ.....	210
Skip mode.....	105
Hochlauf abgeschlossen	188
Dekodierung:Fortsetzung	106
Dekodierung:Stopp	107
Fahrweg:aktuell	89
Zeile:Zähler.....	89
aktiver/folgender:Winkel, zwischen	101
aktuell:Geschwindigkeit:Satzende	101
eingefügt.....	98
Einzelsatzmodus	105
Fahrweg:aktuell	88
Fahrweg:Rest	28
Freigabe Bearbeitung, nächster	106
Kanal	113
Spindel.....	82
Achse:Gantry-Master.....	32
Achse:logische	22, 189
Kanal	23
Werkzeug:einzuwechselndes	148
Werkzeug:Schwesterwerkzeug	148
Werkzeug:Variantenwerkzeug.....	148

O

Offset	
Operation mode	
Orientierung	

OTC

output of command values to real axes

Override

Handbetrieb:zurück fahren:Stopp.....	55
manual mode:move back:stop	54
OTC.....	54
Channel	108
Werkzeug	102
Offset.....	54
radius.....	127
interrupt	117
Eilgangvorschub:Kanal.....	116
Vorschub:Achse	47
Vorschub:Achse:gültig.....	48
Vorschub:Kanal	115
Zeit	124
Zeit:gültig.....	123

P

pAC

Parameter

park

PCS

Platzversatzgruppe

PLC

Position

Positionssollwert

Programmende

Programmzeilennummer

W_AxLinkMasterAxNr	34
X_LinkToLogicalAxis	31
X_LinkToPhysicalAxis	33
Betriebsartwechsel	158
axis	61
Achse:logische Nummer.....	22
Achse:Name	21
Achse:Typ.....	22
Achse:Zustand.....	27
Fahrtweg:NC-Programm:Rest.....	89
Fahrtweg:NC-Satz:Rest.....	28
Offset:Gesamt	25
Offset:Handbetrieb	24
Position:Ist	24
Position:Soll	23
Position:Ziel	23
Referenzpunktfahrt erfolgt	29
Index.....	109
Nachricht an/von.....	209
Nachricht an/von:Inhalt.....	210
Nachricht an/von:Typ.....	210
Reset:Achse	129
Reset:Kanal	130
vorhanden:achsspezifisch	189
vorhanden:kanalspezifisch	187
Achse, in.....	30
Ist:ACS	26
Ist:PCS	24
Messwert:Achse	60
Offset:PCS:Gesamt	25
Offset:PCS:Handbetrieb	24
offset:physical, logical axis	26
Satzvorlauf:Fortsetzung.....	96
Soll:ACS	26
Soll:MCS:Werkzeugmittelpunkt	27
Soll:PCS	23
Ziel:ACS	25
Ziel:PCS	23
Spindel:M19.....	145
erreicht.....	90, 91
H-Funktion	144
M-Funktion.....	144

Q

Quittierung

Technologiefunktion	93
---------------------------	----

R

Referenznocken	50
Referenzposition.....	58
Reaktionsklasse	
Reduktion	
Referenzpunktfahrt	
Regelfenster	
Reglerfreigabe	
Reset	
Restfahrtweg	
Restweg	
Retriggern Tastendruck-Beginn-Ereignis	
Rückwärts	

Rumpftyp	
bei Fehlermeldung.....	194
Geschwindigkeit:Achse	50
Geschwindigkeit:Achse, Zone 1	51
Geschwindigkeit:Achse, Zone 2	51
Geschwindigkeit:Kanal	122
Geschwindigkeit:Kanal, Zone 1	122
Geschwindigkeit:Kanal, Zone 2	123
Löschen	59
Übernahme.....	57
erfolgt:ACS	29
erfolgt:PCS	29
Achse, im.....	29
Kommando	49
Zustand	31
PLC:Achse	129
PLC:Kanal	130
Spindel	81
NC-Satz:PCS	28
Spindel	80
verwerfen:Zustand.....	99
verwerfen:Ende-Marke	126
verwerfen:Kommando	125
Handbetrieb:Durchsetzung eines Tastendrucks	
.....	178
fahren	127
bei Fehlermeldung.....	195

S

Satznummer	31
Schleppfehler	29
simuliertes Vorwärtsfahren	128
Solldrehzahl.....	77
Speicher zum Rückwärtsfahren zurücksetzen	128
Spindelfehler	82
Spindelreset	81
Spindelsteigungsfehler	41
Satzvorlauf	
Satzvorlauf an/aus	
Satzvorschub	
Schwesterwerkzeug	
S-Funktion	
Simulation	
Skip mode	
Sollposition	
Sollwerte	
Spindel	
Spindelbeauftragung	
Spindeldrehrichtung	
Stopp	
Synchronisation	

Bahn:Bewegung, aktuell	97
H-Funktion	143
M-Funktion.....	143
Bahnvorbereitung aktiv	95
Decoder aktiv.....	95
Interpolator aktiv	95
Interpolator:Abstand:Fortsetzposition	96
Interpolator:Zustand	96
an PLC.....	132
programmiert	83
gültig	148
Nummer	148
Achsnnummer	146
Ausführungsdauer, voraussichtlich.....	145
Daten	145
Drehzahl	145
Nummer:Spindelpositionierfunktion	146
Nummer:Spindelschaltfunktion.....	145
Sollwert:Position	145
Bearbeitung EIN/AUS.....	115
NC block	105
erreicht.....	79
ACS	26
MCS:Werkzeugmittelpunkt	27
PCS	23
externe:Achse.....	21
Achse:ACS	22
Auftragsfehler	80
Beauftragung:extern	83
Betriebszustand	78
Drehrichtung	86
Drehzahl:programmiert.....	145
Positionssollwert:M19.....	145
Restfahrweg	80
steht.....	79
Stopp:Programm:Ende	81
extern.....	83
Typ.....	85
Typ.....	86
Bahn:Grund	91
Handbetriebsoffset:zurück fahren.....	55
NC-Programm:Dekodierung	107
Spindel:Programm:Ende	81
satzübergreifend:Technologiefunktion:Kanal ...	137
satzweise:Technologiefunktion:Kanal	136
Technologiefunktion:satzübergreifend:Achse...	135
Technologiefunktion:satzweise:Achse.....	135

T

Temperatureinfluss	42
Tangente	
Tastendruck Beginn/Ende	
Tastenummer	
Technologiefunktion	
T-Funktion	
Tippbetrieb	
Tippgeschwindigkeit	
TwinCAT	
Typ	

Bahn	102
Handbetrieb:Durchsetzung eines Tastendrucks	177
Handbetrieb:Durchsetzung eines Tastendrucks	177
Handbetrieb:Eilganggeschwindigkeit während Tastendruck	180
Achse:Daten	138
Anzahl	135, 136, 137
Anzahl:nicht quittiert:satzübergreifend	144, 146, 147
H-Funktion	143
Kanal:Daten	140
M-Funktion	143
Quittierung	93
S-Funktion	145
Synchronisation:satzübergreifend:Achse	135
Synchronisation:satzübergreifend:Kanal	137
Synchronisation:satzweise:Achse	135
Synchronisation:satzweise:Kanal	136
T-Funktion	147
Werkzeug:Identifikation	147
Werkzeugeinheit:Information	147
Daten	147
Achsnummer	170
Geschwindigkeit	170
Parametrierung	169
Verfahrgeschwindigkeit	163
Handbetrieb	170
Box	44
Gerät:Identifikation	44
Nachricht:an/von PLC	210

U

Umkehrspanne	42
Updatezyklus	
HLI:achsspezifisch	190
HLI:kanalspezifisch	188

V

V.E.-Variable	149
Vorsteuerung	42
Variable	
Variantenwerkzeug	
Vektor	
Verfahrgeschwindigkeit	
Verfahrweg	
Verschleiß	
Versionsbezeichnung CNC	
Versionskennung	
Verweilzeit	
Vorschub	
Vorschubfreigabe	
Vorschubstopp EIN/AUS	

extern	149
gültig	148
Nummer	148
Bahn:Tangente	103
Bahn:Tangente:Normal	103
Fläche:Normal	103
Werkzeug:Richtung	102
Jogbetrieb	163
Tippbetrieb	163
Jogbetrieb	164
Werkzeug:Radius	127
bei Fehlermeldung	194
CNC ® PLC	187
PLC ® CNC	186
aktiv	94
Bahn:Grenzwert unterschritten	96
Override:Achse	47
Override:Achse:gültig	48
Override:Kanal	115
Achse	45
Achse	46
Kanal	110

W

Wahlweiser Halt	114
Wartezeit	
Watchdog	
Werkzeug	
Winkel	
Ecke	100
CNC überwacht PLC	182
PLC überwacht CNC	184
Anzahl:nicht quittiert:satzübergreifend	147
Einheit:Information	147
Geschwindigkeit:minimale:Ignorierung	53
gültig:Schwesterwerkzeug	148
gültig:Variantenwerkzeug	148
Identifikation	147
Korrektur:Online:Radius	127
Nummer:einzuwechselndes	148
Nummer:Schwesterwerkzeug	148
Nummer:Variantenwerkzeug	148
Orientierung	102
Richtung:Vektor:	102
Standgröße:Erfassung, ausschalten	125
aktiver/folgender NC-Satz	101

Z

Zeitoverride	124
Zähler	
Zeiger	
Zeile	
Zeit	
Zeitangabe	
Zielposition	
Zielzustand	
Zusatzinformation	
Zustand	
Zustands-Istwert	

Zeile:NC-Programm	89
HLI	211
Nummer:Fehlermeldung.....	191
Nummer:Fehlermeldung:Dienstfunktion	192
Zähler:NC-Programm	89
Ecke	100
bei Fehlermeldung:Interruptzyklen	193
bei Fehlermeldung:Interruptzyklen seit Systemstart	193
Bewegung:G01/G02	99
gültig.....	123
ACS	25
PCS	23
Spindel	78
Betriebsart:Betriebsartwechsel.....	157
H-Funktion	144
M-Funktion	144
Achse:Interpolator	33
Antrieb	38
Handbetrieb	162
:Betriebsart	159

16 Appendix

16.1 Suggestions, corrections and the latest documentation

Did you find any errors? Do you have any suggestions or constructive criticism? Then please contact us at documentation@isg-stuttgart.de.

The latest documentation is posted in our Online Help (DE/EN):



QR code link: <https://www.isg-stuttgart.de/documentation-kernel/>

The link above forwards you to:

<https://www.isg-stuttgart.de/fileadmin/kernel/kernel-html/index.html>



Notice

Change options for favourite links in your browser;

Technical changes to the website layout concerning folder paths or a change in the HTML framework and therefore the link structure cannot be excluded.

We recommend you to save the above "QR code link" as your primary favourite link.

PDFs for download:

DE:

<https://www.isg-stuttgart.de/produkte/softwareprodukte/isg-kernel/dokumente-und-downloads>

EN:

<https://www.isg-stuttgart.de/en/products/softwareproducts/isg-kernel/documents-and-downloads>

Email: documentation@isg-stuttgart.de



© Copyright
ISG Industrielle Steuerungstechnik GmbH
STEP, Gropiusplatz 10
D-70563 Stuttgart
All rights reserved
www.isg-stuttgart.de
support@isg-stuttgart.de

